

Deins Kovalcuks 1847031
Zofia Rynkiewicz 1817760
Theodore Visoiu 1791346
Maxim Yordanov 1812459

Preface

Before diving into the main topic of our research, we spent some time conducting background research to understand what had already been studied in the field of underwater wireless communication. This helped us identify any gaps in the existing research and determine where we could contribute something new.

During this process, we noticed that while many communication methods have been tested in underwater environments, the IEEE 802.11ah standard, also known as Wi-Fi HaLow, has not been explored much in this context. HaLow is known for its low power consumption, long-range, and high data rate capabilities, which could make it a good option for underwater use. However, we found no studies testing its performance under such conditions.

One recent paper we found [1] suggests there is growing interest in HaLow for unique and challenging environments, but it still has not been tested underwater. Because of this, we therefore decided to investigate the performance of 802.11ah in underwater settings.

Accordingly, we focused our project on testing and evaluating how this protocol behaves in underwater communication scenarios and investigate its properties with the use of the ns-3 simulator [2].

Acknowledgments

We would like to express our deep gratitude to [Nirvana Meratnia](#) for her warm welcome into research, the critical viewpoints that kept our work and thinking sharp, and the enthusiasm she shared for our discoveries. Our thanks also go to [Richard Verhoeven](#) for the many days (and countless coffees) spent discussing our results, rigorously questioning their validity, and patiently enduring our unconventional server usage patterns.

Contents

1	Introduction	1
2	Related Work	2
2.1	Peer-to-peer scenario	2
2.2	Sensors-to-AP scenario	3
2.3	Data muling scenario	3
3	Scenarios	4
3.1	Peer-to-peer	4
3.2	Sensors-to-AP	5
3.3	Data mulling	5
4	Simulation environment	6
4.1	ns-3 simulator	6
4.2	Challenges of using ns-3	6
4.3	Model for underwater communication with EM signals	7
5	Simulation setup	9
5.1	Basic setup and simulation parameters for the scenarios	9
5.2	Evaluation metrics	11
6	Simulation results and discussion	12
6.1	Peer-to-peer	12
6.2	Sensors-to-AP	23
6.3	Data mulling	31
7	Conclusions and future work	38
A	Propagation model formulas	40
B	RAW configuration results	41
C	Individual contributions	43
C.1	Deins Kovalcuks	43
C.2	Zofia Rynkiewicz	43
C.3	Theodore Visoiu	43
C.4	Maxim Yordanov	43
References		44

1 | Introduction

With the advancement of Internet of Things (IoT) technologies, underwater wireless communication has gained significant attention for applications such as oceanographic research, environmental monitoring, control of underwater pipelines, and military operations [3]. Over the past decades, researchers have explored underwater wireless communication using optical, acoustic, and electromagnetic (EM) signals, each with its own advantages and limitations [4].

Acoustic signals can travel long distances (up to several kilometers), making them suitable for large-scale deployments. However, acoustic links exhibit low data rates and high latency, rendering them unsuitable for real-time monitoring applications [4], [5].

Optical links, while offering high bandwidth at shorter distances (up to approximately 100 meters), are highly sensitive to water turbidity and require precise line-of-sight alignment between the transmitter and receiver, limiting their practicality [4], [5].

EM waves, particularly in the radio frequency (RF) range, present a promising alternative. Unlike optical and acoustic signals, RF waves are independent of water turbidity and allow for more flexible transmitter and receiver orientations. Furthermore, the EM waves can easily cross water-to-air or water-to-earth boundaries, following the path of least resistance, which is neither the case for acoustic nor optical communication [6].

They also offer high bandwidth and low latency. However, due to the high attenuation of EM signals in water, their effective range is limited to short distances, making them best suited for deployments with small gaps between the nodes [5], [7]. The distance that the RF signal can travel depends on conductivity and frequency. The higher the conductivity and operating frequency, the more the signal gets attenuated [8].

In [8], the maximum measured and simulated range of 802.11af (White-Fi) operating with a frequency of 768 MHz was around 200 cm. The conductivity of water was around 0.05 S/m, which is at the classification boundary for freshwater [8].

HaLow achieves higher data rates than comparable IoT protocols such as LoRa, NB-IoT, and SigFox [9]. Moreover, HaLow outperforms other IEEE 802.11 variants (e.g. Wi-Fi, White-Fi) in power efficiency [10]. Maudet et al. [11] identify HaLow as one of the most promising technologies for scalable, high-performance, energy-efficient IoT deployments.

This study evaluates the performance of Wi-Fi HaLow underwater using ns-3 simulations. We model the following scenarios to determine its feasibility for practical applications like wireless subsea connectors, AUV underwater docking stations [12], [8], [4] and underwater wireless sensor networks [13], [14], [15]. To gain deeper insights into its benefits and limitations, we will simulate three different scenarios with varying node allocations and in different environmental conditions. The peer-to-peer simulation scenario is meant to simulate wireless subsea connectors and AUV underwater docking stations. Sensors-to-AP and Data Muling simulation scenarios are meant to simulate underwater wireless sensor networks. For each scenario, the simulation results include throughput, latency, power consumption, and energy efficiency.

2 | Related Work

2.1 | Peer-to-peer scenario

[8] is a publication that closely resembles our simulation setup for peer-to-peer simulation. The main differences are:

1. Our simulations operate at 900 MHz, whereas [8] explores simulations with 768 MHz, 2.462 GHz and 5.240 GHz signals.
2. We fix the transport layer protocol in our simulations to TCP, whereas [8] considers both TCP and UDP protocols.
3. We vary the temperature 0°C - 30°C , whereas [8] sets it to 22.5°C.
4. We explore the scenarios with different salinities (0.01, 0.5 and 35 ppt), whereas [8] fixes the conductivity to 0.5, which corresponds to the salinity of 0.29 ppt (Equation (A.8)).
5. Authors in [8] verify their simulation results experimentally, but we do not.
6. [8] only measures throughput, whereas we, apart from throughput, also measure latency, power consumption, and energy efficiency.
7. We use link layer protocol 802.11ah (WiFi Halow), whereas [8] uses 802.11g.

Despite the differences, we expect comparable results for peer-to-peer simulations with freshwater salinities, TCP protocol and the temperature of 22.5°C. The study reported the signal range of around 200 cm using the 768 MHz frequency, with the throughput reaching around 21 Mbits/s. We expect the range in our simulation to be slightly smaller, as HaLow operates on 900 MHz compared to the 768 MHz in [8]. Since the channel widths of Halow in our simulations are 1 and 2 MHz, while [8] (implicitly) simulates 20 MHz channels, as specified by 802.11g, we expect our maximum throughput to be much smaller.

The study in [12] is also similar to our peer-to-peer scenario setup: The salinity is fixed at 33 ppt, and the temperature is set to 20°C. However, the investigated frequencies are 4.7 GHz and 5.7 GHz, and the only metric used is the received power with the initial transmission power of 0 dBm. The estimated signal range is in the neighbourhood of 5 to 7 cm. We expect to have a longer range for 35 ppt and 20°C because we investigate a frequency that is 5 times lower; hence, the attenuation should be lower as well. The link layer protocol is not specified. Furthermore, [12] uses an optimized antenna, compared to the omnidirectional one used throughout our HaLow simulations.

Researchers in [4] conducted peer-to-peer freshwater and saltwater experiments with the IEEE 802.11ac standard using the 5.2 GHz frequency and obtained results consistent with the simulations in [12].

In [16], the signal range is estimated to be around 2.4 meters in fresh water and 0.25 m in water with a salinity of 5 ppt for the frequency 700 MHz. Water temperature is not specified.

The results from the experiments in [17] show that for 2.4 GHz frequency, using the 802.11 b/g standard, at water temperature of 20 °C and in fresh water, the maximum achievable range is 26 cm.

[18] conducts peer-to-peer experiments with LoRa technology at the frequency of 868 MHz. The orientation of the nodes in space is vertical, which makes it more similar to the Sensors-to-AP scenario. The maximum

range is not specified, but it is at least 1.4 meters based on Received Signal Strength Indicator (RSSI) data. The properties of the water were not specified.

Similarly, [19] explores communication properties of LoRa at 433 MHz in an experimental peer-to-peer scenario. It shows that, for two horizontally oriented nodes in space, as depth from the water surface increases, the received signal strength worsens. Here, the range of the signal is around 1.5 m. The properties of the water were not specified.

2.2 | Sensors-to-AP scenario

The paper [20] proposes an underwater sensor network (UWSN) architecture with a buoy at the water surface and sensor nodes on the seabed. Sensor nodes would transmit data using an RF signal to the intermediate nodes, which would amplify it and transmit it to the buoy. Our Sensors-to-AP scenario is similar, except that it does not have intermediate nodes, and the layout of the nodes on the seafloor is fixed to a 2D grid.

[6] also describes the results of a case study with a similar implementation to what was presented in [20]. The results consist of the analysis of propagation delay for different packet sizes and power consumption.

[21] compares the network lifetime for UWSNs, where nodes are placed in a 2D grid on the seafloor and there is a buoy on the water surface acting as a sink. Using simulations, the authors analyze and compare the network lifetime when the nodes directly relay data to the sink, when there is a dedicated fixed cluster node and when the cluster node role can be adapted by other nodes dynamically. The first situation is the one we explore in the sensors-to-AP scenario.

[13] describes capabilities, limitations and applications of RF communication in shallow water networks. The author mentions that the network architecture found in the sensor-to-AP scenario is common in such settings.

[22] simulates a UWSN similar to the Sensors-to-AP scenario, but with intermediate nodes. The frequency of the simulated signal is 3 MHz, which is much smaller than the frequency of the signal in our simulations. It claims that throughput in bit/sec and energy efficiency in bit/Joule are the main performance metrics for UWSNs.

2.3 | Data muling scenario

[23] describes a framework in which data mules are used to implement a communication channel between the sensors on the seafloor and a central station on the surface. The authors also conducted a physical experiment, which showed that this system could achieve 150 times higher throughput compared to acoustic communication. This enables applications such as video streaming.

Throughout the literature analysis phase, a significant number of papers have been uncovered pertaining to the analysis of the optimal path traversed by the data mule [14], [24]–[28].

However, to the best of the authors' knowledge, there is no publication assessing the HaLoW protocol in data muling scenario.

This work fills this gap, with the aim of investigating the performance of the IEEE 802.11ah standard for such systems.

3 | Scenarios

To evaluate HaLow's feasibility in different underwater applications, we analyze three distinct node allocation scenarios. They differ in terms of the number of deployed stations, placement, and access point (AP) location. This yields varying interference levels, connectivity dynamics, and overall scalability.

In each simulation scenario, HaLow's performance will be evaluated in both freshwater and seawater environments, as they present properties affecting signal propagation. In all scenarios, it is assumed that the sensors are deployed in open sea conditions, away from shallow waters, where coastal features such as cliffs do not impact signal propagation. Across all scenarios we assume that the sensors and the buoy remain stationary throughout the operation. We only consider the signal propagation through homogeneous water, without considering medium changes (from water to air and vice versa) and ground wave reflection interference patterns.

3.1 | Peer-to-peer

This scenario investigates the communication between two nodes - a transmitter (station) and a receiver (access point) - where they exclusively exchange data (see Figure 3.1 for an overview). The primary objective is to evaluate the protocol's performance at varying distances, water temperatures, and salinities. Since the nodes may be deployed in diverse aquatic environments where temperature fluctuations are significant, this investigation aims to assess the impact of temperature on communication effectiveness. For consistency, salinity is assumed to be constant for each type of water considered in the study.

Moreover, we set the RAW configuration to contain only a single RAW group with only one slot for the transmitting node. We investigate 1 MHz and 2 MHz channel widths. For the 1 MHz channel width we picked a moderate data rate using Orthogonal Frequency Division Multiplexing (OFDM) with a coding rate of $\frac{1}{2}$, typical for long-range low-data-rate IoT applications. For the 2 MHz channel width we picked the highest data rate available using OFDM with a coding rate of $\frac{7}{8}$, as to investigate short-distance high-throughput communication scenarios.

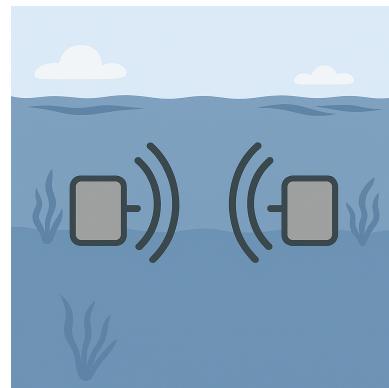


Figure 3.1: Peer-to-peer communication.

Image generated using ChatGPT (DALL·E) [29].

3.2 | Sensors-to-AP

This scenario explores an underwater network topology where multiple sensor nodes are deployed on the seafloor, collectively forming a network that communicates with a single access point (AP) located on a buoy at the water's surface [13], as illustrated in Figure 3.2. In this study, we assume that the AP's receiving antenna is submerged underwater. Once the signal reaches the AP, it can be efficiently relayed to a shore station for further processing. This data relay is outside the scope of our study.

A key focus of this scenario is to assess the impact of interference on signal transmission from multiple nodes to a single AP, distinguishing it from direct peer-to-peer communication.

As we are also investigating the influence of interference in this scenario, we will additionally examine the restricted access window (RAW) configuration, which allows for the best performance metrics.

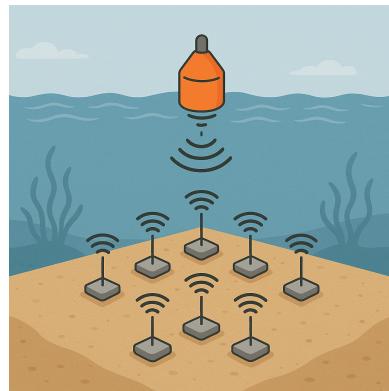


Figure 3.2: Sensors-to-AP communication.

Image generated using ChatGPT (DALL·E) [29].

3.3 | Data mulling

In this scenario, we investigate a node allocation strategy in which multiple sensor nodes are deployed on the seafloor, while an Autonomous Underwater Vehicle (AUV) functions as a mobile Access Point (AP). The AUV traverses the network (presented in Figure 3.3), collecting data from static sensor nodes to facilitate efficient communication. Sensors connect only when the AUV is in range to facilitate data transfer.

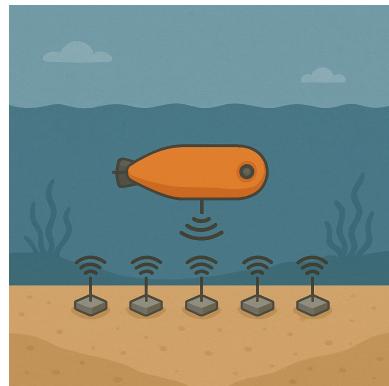


Figure 3.3: Data mulling. AP is an AUV collecting data from the sensors.

Image generated using ChatGPT (DALL·E) [29].

4 | Simulation environment

4.1 | ns-3 simulator

For simulating 802.11ah behavior underwater, we utilized the open-source ns-3 simulator [2]. Our research was based on [30]’s version of the ns-3 simulator, which incorporated WiFi HaLow support into ns-3.23. The code for our simulations was further influenced by [31]’s simulations, which served as a foundation. In our implementation, we introduced novel path loss and propagation delay models, which are described in Section 4.3.

We leverage standard ns-3 modules: `YansWifiPhy` for the physical layer and our derived `StaWifiMac`/`ApWifiMac` for the MAC layer. For traffic, we use the default TCP/IP stack with `BulkSendApplication` and `PacketSink` for performance analysis [2]. We modified `ApWifiMac` to enable station re-connection, which was a prerequisite for the AUV scenario Section 3.3. These modifications entailed removing testing code from [30] in the association routines and replacing it with checks and internal state resets that enable a station to correctly re-associate with the AP. For power analysis, we utilized ns-3’s Energy Module [32] to provide each node with a “battery” that depreciates during transmissions, enabling power usage studies. The current draw constants

$$I_{tx} = 0.380 \quad I_{rx} = 0.313 \quad I_{idle} = 0.273 \quad I_{sleep} = 0.033$$

were provided by [33]. We extensively utilized ns-3’s tracing mechanism for logging purposes, allowing us to post-process and visualize the data for each combination of the simulation’s parameters (conductivity, node spacing, etc.) in Python.

4.2 | Challenges of using ns-3

While working on this paper, we promptly recognized that ns-3’s extensibility and customizability were both its greatest strengths and weaknesses. This is primarily due to the lack of backward and forward compatibility between modules, as well as the simulator’s encouragement to modify its source code directly, without the ability to effectively migrate those changes to later versions. Consequently, we were compelled to utilize the ns-3.23-3.25 hybrid developed by [30], as implementing HaLow support in a later version of ns-3 warranted a separate paper. These limitations subsequently cascaded into host operating system (OS) compatibility, dependency installations, and so on. We successfully executed the simulation on Ubuntu 20.04 and 22.04 with minimal modifications to the workflow. However, newer OS versions encountered significant challenges in running the Python scripts provided by ns-3 due to the upgrade of the OS-level Python version from 2 to 3. Another notable issue was the absence of comprehensive support and documentation for older versions of ns-3, resulting in the inaccessibility of several logging tools, utilities, and helpers. Consequently, we had to implement them ourselves.

A potential concern arises regarding the accuracy of our simulation. Later versions of ns-3 introduced superior models for the physical layers (such as `SpectrumWifiPhy` in ns-3.26), additional TCP congestion control variants (in ns-3.26), new error models (in ns-3.33), and new antennas (in ns-3.34). All of the above had the potential to enhance the precision of our results had they been available for our use. Furthermore, the WiFi HaLow implementation we based our study on lacks support for all 802.11ah channel widths and data rates, limiting our testing to the 1 MHz and 2 MHz wide bandwidths. Therefore, we recommend further investigations into the performance of HaLow underwater for channel widths ranging from 4 to 16 MHz.

Furthermore, for the graphs in Section 6.2, we tried to ”split” the total power consumption as reported by ns-3’s Energy module by tracking the state changes and cross-referencing them with the power draw as it

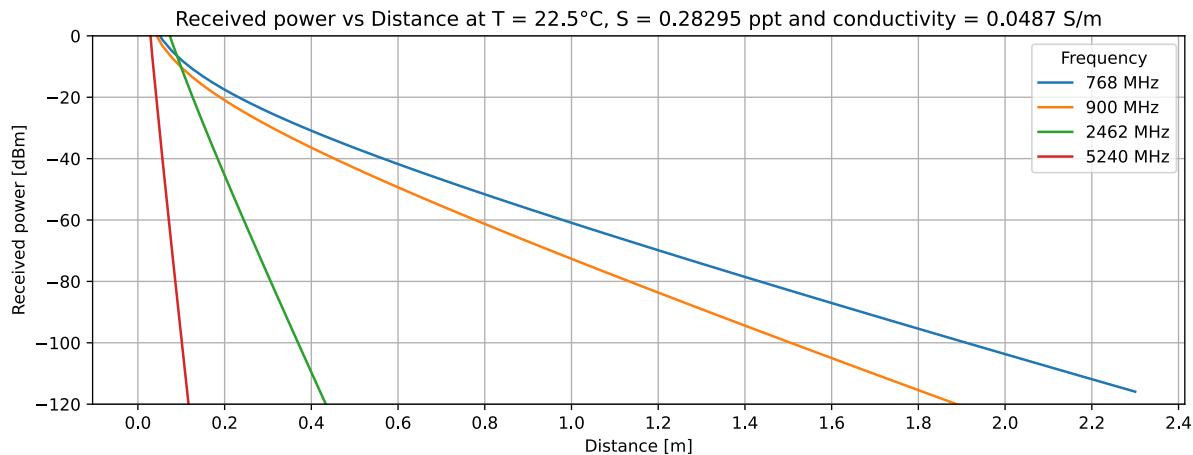
happens. Unfortunately, we have found that the state switches and/or power draw are not always reported properly by ns-3, causing race conditions and a mismeasurement in the per-state split power graphs.

4.3 | Model for underwater communication with EM signals

Underwater electromagnetic (EM) signals differ significantly from those in air due to the distinct electrical properties of water, namely conductivity (σ), permeability (μ), and permittivity (ϵ), all of which influence signal propagation. In particular, higher conductivity, which varies with the presence of ions, results in greater signal attenuation. Permeability, which determines a medium's ability to store magnetic energy, remains unchanged in water compared to free space, as water is a non-magnetic medium. Conversely, permittivity depends on the medium, as it defines the ability of the material to transmit an electric field. To accurately define the model for this complex underwater environment, we followed the formulas provided in [20], [8], which describe the radio frequency path loss in underwater wireless sensor networks due to signal attenuation. This approach allowed us to base our model on salinity and temperature, as other relevant parameters can be expressed in terms of these two variables. To compute the total path loss, we incorporated the Friis transmission equation [34], assuming an ideal isotropic antenna with no thermal loss, to account for the spreading loss - the reduction in signal power due to the geometric dispersion of the radio frequency signal.

In the incipient stages of our research we also considered using Aqua-Sim NG [35]. We later decided to pivot away as Aqua-Sim NG is designed for acoustic instead of electromagnetic waves.

After setting up the model, we compared its performance to the results shown in [8] (Section 4.3). However, it was difficult to replicate the exact conditions, as the authors did not specify them explicitly. Below, we present the performance of our model, including results for a carrier frequency of 900 MHz.



4.3.1 | Path loss

We define the path loss with the following formula:

$$L_{pathloss}[dB] = 20 \Re(\gamma) D \log_{10} e + 20 \log_{10}\left(\frac{4\pi f D}{c}\right) \quad (4.1)$$

where the D is the distance between the transmitting and receiving nodes, carrier frequency $f = 900\text{MHz}$, and the speed of light $c = 300'000'000 \text{ m/s}$ and propagation constant γ of a sinusoidal electromagnetic wave is given by:

$$\gamma = \sqrt{j\omega\mu_0(\sigma + j\omega\varepsilon)} \quad (4.2)$$

where ω is the angular frequency of the signal, $\mu_0 = 4\pi \times 10^{-7}$ is the permeability of water and ε is the absolute permittivity, which can be calculated by $\varepsilon = \varepsilon_0 \varepsilon_r$, where $\varepsilon_0 = 8.85 \times 10^{-12} F/m$ is the permittivity in free space and ε_r is the relative permittivity, which we calculate by Equation (A.1) and conductivity σ is also calculated by Equation (A.8).

4.3.2 | Final formula for received power

Putting everything together, the formula for the received power at the reception sensor is calculated by [8]:

$$P_{rx}[dBm] = P_{tx} + G_{tx} + G_{rx} - L_{pathloss} \quad (4.3)$$

We used transmission power $P_{tx} = 0$, transmitter antenna gain $G_{tx} = 1$, receiver antenna gain $G_{rx} = 1$.

4.3.3 | Propagation Delay

For determining the appropriate propagation delay we used the following formula, according to the definition:

$$t_{delay} = \frac{D}{v_p} = \frac{D\beta}{\omega}. \quad (4.4)$$

where the phase constant β is the imaginary part of γ described in Equation (4.2):

$$\beta = \Im(\gamma) \quad (4.5)$$

5 | Simulation setup

5.1 | Basic setup and simulation parameters for the scenarios

Our simulated AP runs at 900 MHz and each sensor transmits data continuously, with a packet payload size of 64 bytes. For the data points with 1 MHz-wide channels we are using the `OfdmRate1_2MbpsBW1MHz` data rate, and for 2 MHz-wide channels we are use the `OfdmRate7_8MbpsBW2MHz` data rate. We considered using adaptive data rate algorithms (for instance Minstrel, defined at [36]), but decided against it to avoid adding complexity to the simulations and introducing additional variability that could affect the results.

5.1.1 | Peer-to-peer

In this scenario, simulations were conducted using 1 MHz and 2 MHz-wide channels across three representative water salinities - 0.01, 0.5, and 35 parts per thousand (ppt). These salinity levels were selected based on their prevalence in Earth's oceans and rivers, as reported by [37], [38], and [39]. Across the simulations, the temperature of water and sensor spacing is varied.

5.1.2 | Sensors-to-AP

In this scenario, we vary the number of sensors deployed on the seabed, ensuring that all nodes are within range and capable of reliable communication with the access point (AP). The number of sensors and the distance between the seabed and the buoy (access point) is varied across the simulations. The simulations use 1 MHz-wide channels and a salinity of 0.01 ppt (representing freshwater), as both parameters support longer communication ranges, as concluded from the peer-to-peer scenario. The water temperature was set to 20 °C, following the studies [12], [40], [41]. Moreover, we investigated two crucial aspects: optimal restricted access window (RAW) group configuration and and the selection of a fixed sensor deployment area on the seabed, which had to be determined in advance.

1. **Area of the deployment for the sensors:** Before the deployment of the sensors, we closely investigated the behavior of the communication between the nodes in the peer-to-peer scenario which allowed us to determine the largest distance between the nodes for which the communication is still reliable in the temperature of 20 °C, which turned out to be 1.3 m. Using this information, and given that the layout of the nodes is fixed to a square grid, we computed the maximum area of deployment without the hidden terminal problem. Next, we computed the max height of the AP such that all deployed nodes in all scenarios would be in range. The dimensions of this configuration are illustrated in Figure 5.1. The maximum distance from the edge of the square to the tip of the cone is 1.3 m, which corresponds to the longest distance at which reliable communication was achieved (investigated in the peer-to-peer scenario). To eliminate the hidden node problem, the diagonal of the square is also set to 1.3 m. Under these constraints, the maximum height of the cone, representing the buoy's position on the surface, was determined to be 1.13 m.

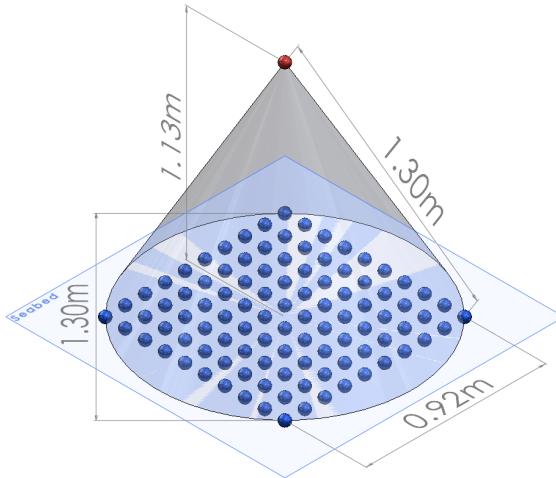


Figure 5.1: Dimensions of the cone used for the area calculations on which the sensors are deployed.

Image generated using SolidWorks [42].

2. **Allocation of the nodes within the square area:** As the number of sensors varies among the simulations, we decided to allocate them within the square area in the following way. The positioning proceeds by first calculating the smallest square grid that can accommodate the given number of sensors. For instance, if there are 10 sensors, they will be organized into a 4×4 grid, since $\sqrt{10} \approx 3.16$, and rounding up yields 4. Each sensor is placed along evenly spaced rows and columns within the square, with distances between neighboring sensors adjusted so that the total width and height remain within the 0.92 m limit; the maximum side length of a square that has a 1.3 m diagonal, therefore avoiding the hidden node problem. If there are not enough sensors left over to fill the last row, the last row will be left partially empty. This ensures a uniform distribution across the area, maximizing inter-sensor spacing while keeping all nodes within the designated square.
3. **RAW configuration:** For this analysis, we deployed 100 nodes in a 10×10 configuration as explained above. We then tested all RAW configurations defined in the following set, assuming a beacon interval of 102 400 μ s and no page slicing. Furthermore, we also disabled transmissions across slot boundaries.

$$\{(N_{groups}, N_{slots}) \mid N_{groups}, N_{slots} \in [1, 100] \wedge N_{groups} \cdot N_{slots} \leq 100\}$$

Afterwards, we evaluated the performance of the RAW groups based on the mean throughput, latency, power usage, and energy efficiency. The metrics were computed per node, and then all per-node statistics were averaged for each configuration to determine the performance metrics of each RAW configuration. The results are presented in table Table B.1. We picked the one RAW group with one slot configuration as the baseline configuration for all further testing whenever there is more than one sensor on the network.

5.1.3 | Data mulling

In this scenario, a line of 30 sensors is deployed on the seabed, and we have an autonomous underwater vehicle (AUV) going back and forth 0.5 m above the sensors. The AUV starts at one end of the sensor line, goes to the other end, turns around, and comes back. It repeats this behavior until the simulation finishes (after 15 min for non-wake-up sensor simulations and 10 min for wake-up sensor simulations). The speed of the AUV goes up to 2 m/s and remains constant throughout the simulation. This upper bound

on the speed was picked because it is the current operational limit of most AUV offerings, as discussed by [43] and [44]. The spacing between the sensors and the AUV speed is varied across the simulations. The simulations use 1 MHz-wide channels and a salinity of 0.01 ppt (representing freshwater), as both parameters support longer communication ranges, as concluded from the peer-to-peer scenario. The water temperature was set to 20 °C, following the studies [12], [40], [41]. Finally, in order to investigate an improvement of the power usage and energy efficiency a 'smart wake-up sensor' functionality was introduced. This 'sensor' wakes up the nodes when the AUV is in range, after a state of 'deep sleep' which is more energy efficient than the sleep state the modes would need to be in if they could not rely on the 'sensors' and thus saving additional power. The real-life implementation of those 'sensors' is not in the scope of this report but it could be based on light or acoustic signals.

5.2 | Evaluation metrics

To evaluate the performance of the protocol, we decided to investigate the throughput, latency, power usage, and energy efficiency for each scenario. We only take into account the traffic going from the node to the access point. Below we provide a description on how these metrics are calculated using the simulation.

5.2.1 | Latency

This metric is calculated using ns-3's [2] FlowMonitor's capability to track the packets exchanged within the network. For each flow, the cumulative end-to-end delay of all received packets (`delaySum`) is divided by the number of received packets (`rxPackets`) to obtain the mean latency per flow. If the scenario includes multiple transmitters (multiple stations), the average latency is then computed across all flows to produce a single representative latency value for the scenario.

5.2.2 | Throughput

We assess throughput also with the use of FlowMonitor [2]. The cumulative sum of the received bytes in a flow (`rxBytes`) is divided by the difference (`timeLastRxPacket - timeFirstTxPacket`), so the time when the last packet of with a particular ID was received and first packet with this id was transmitted by the node. If the scenario includes multiple transmitters (multiple stations), the average throughput is then computed across all flows to produce a single representative value for this metric for the scenario. The result is also scaled to the megabits per second [Mbps] unit.

5.2.3 | Power usage

Power usage per node is tracked using ns-3's Energy model [2]. Each node is initialized with an energy budget of 100'000'000 J provided by an ideal battery and is assigned constant current values for each mode as reported in [33]:

$$I_{tx} = 0.380 \quad I_{rx} = 0.313 \quad I_{idle} = 0.273 \quad I_{sleep} = 0.033$$

A constant supply voltage of 3V is assumed. To simulate radio deactivation, all current factors are temporarily set to zero; reactivation restores the original values. The results (in Jules) are then post-processed by dividing them by the simulation's runtime to determine the power usage in Watts.

5.2.4 | Energy efficiency

To compute energy efficiency, the total number of received bytes (`rxBytes`) obtained from `FlowMonitor` is divided by the energy consumed across the lifetime of the node. In scenarios involving multiple transmitters

(stations), the average energy efficiency is calculated across all flows, resulting in a single representative metric for the scenario instance.

6 | Simulation results and discussion

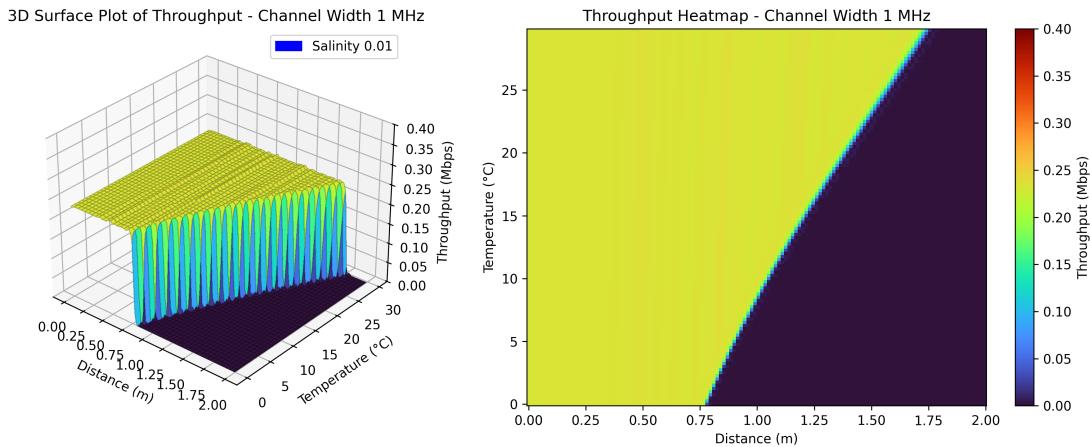
In the following section, we present the results obtained after running the ns-3 simulator with the corresponding setup for the three different scenarios. For each scenario, we present a set of 3D graphs, covering various performance metrics: latency (s) computed for successfully transmitted packets, throughput (Mbps), power usage (W) and power efficiency (Mb/J). The metrics are plotted against crucial parameters that affect the performance:

- In the Peer-to-peer scenario: temperature ($^{\circ}\text{C}$) and distance between the sensors (m).
- In the Sensors-to-AP scenario: sensor count on the seabed (#) and sensor depth (m).
- In the Data mulling scenario: speed of the AUV (m/s) and the sensor spacing (m).

Each 3D graph is also paired with the corresponding heatmap, which helps with data analysis. The description and discussion of the results follow each pair of graphs.

6.1 | Peer-to-peer

6.1.1 | Throughput



(a) Channel width 1 MHz, Salinity 0.01 ppt

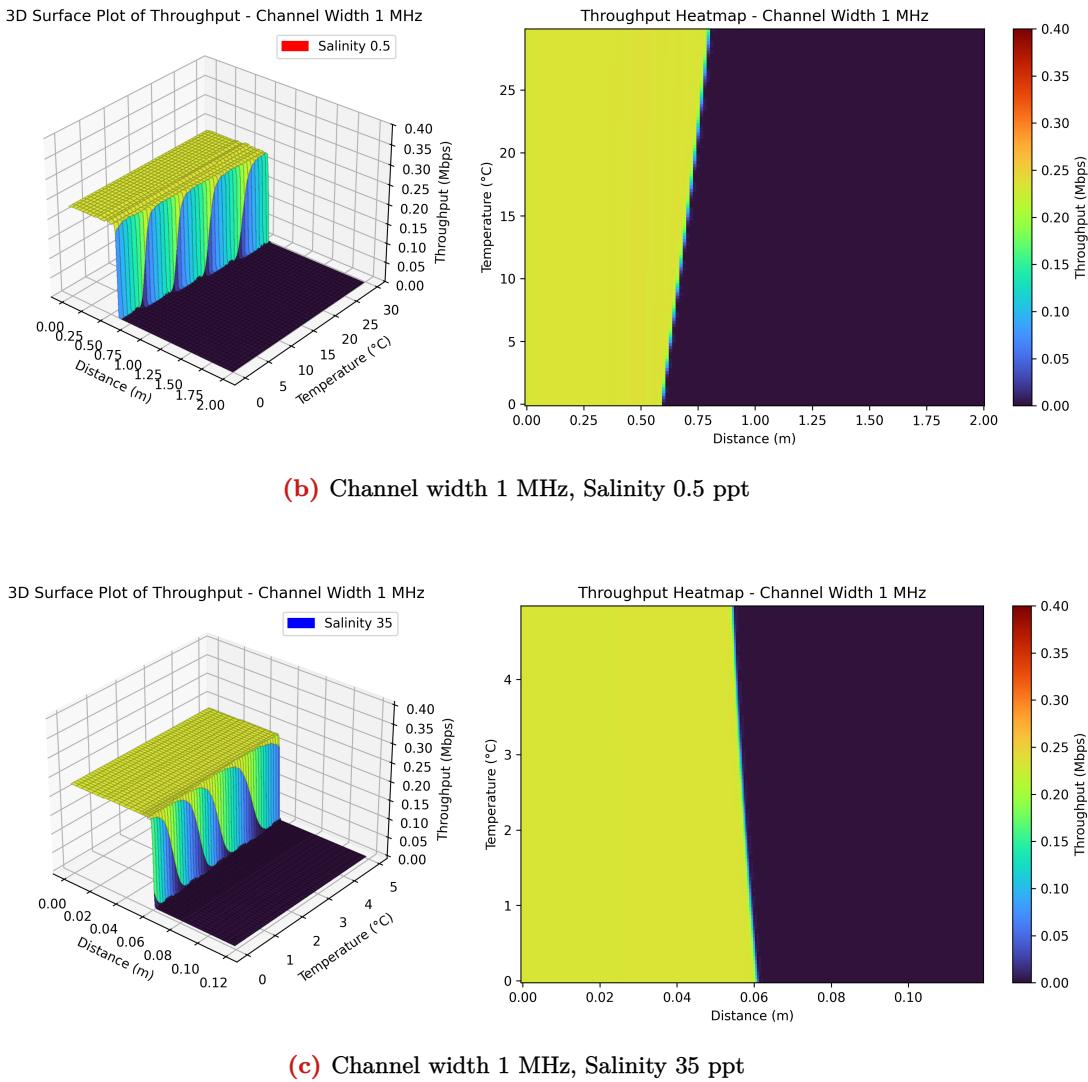
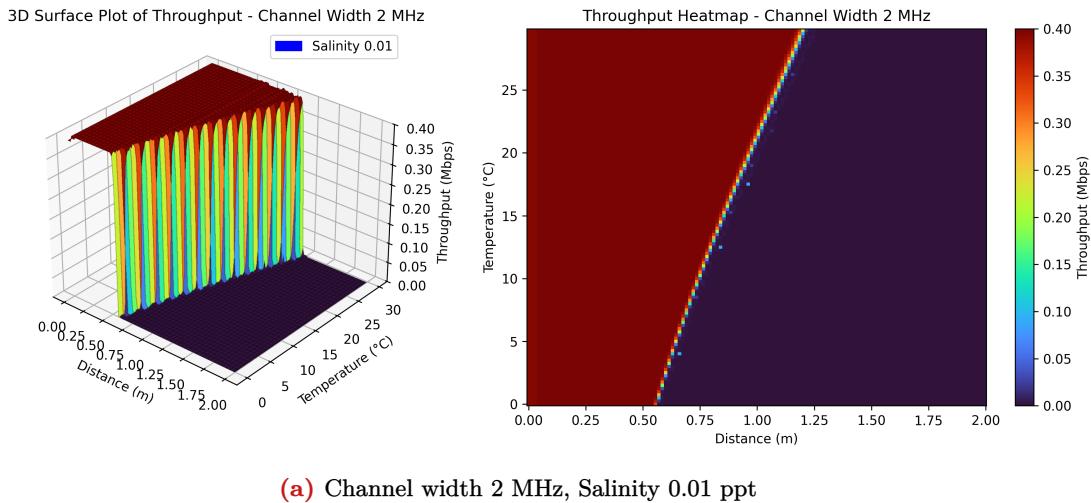


Figure 6.1: Mean throughput STA to AP vs. distance and temperature for 1 MHz channel in the peer-to-peer scenario

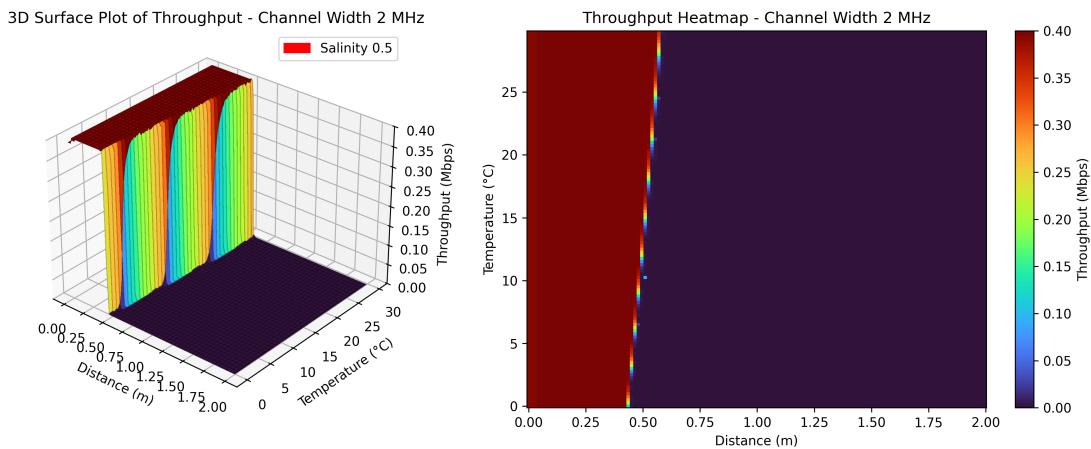
In the peer-to-peer scenario, for 1 MHz-wide channels, the throughput reaches around 0.2 Mbps, and then drops at a certain temperature and distance between the nodes that exchange data. The maximum throughput, as expected, is smaller than in [8]. At a salinity of 0.01 ppt (Figure 6.1a), this change can be noticeable from 0.75 m and temperature around 0 °C, and for higher temperatures, this drop is observed for higher distances between the nodes.[45] reported a similar pattern for a signal of frequency 1 GHz in freshwater, where the attenuation factor (that describes how much the signal weakens when traveling underwater) increases for lower temperatures. The estimated values given were 2.7 (dB/cm) in 5 °C and 2.3 (dB/cm) in 22 °C. For temperature 22.5°C, the signal range is around 1.75 m, which is consistent with the results in [8]. Similar patterns can also be seen for salinity 0.5 ppt (Figure 6.1b), but the cut-off at 0°C is visible when the distance between the nodes is around 0.6 m. The effective signal range between the nodes does not increase significantly with the increase of temperature - the maximum distance between the nodes with non-zero throughput is around 0.75 m, regardless of temperature.

On the other hand, the inverse behavior, where higher temperature increases attenuation, can be seen in Figure 6.1c, which shows the results for salinity of 35 ppt. For this salinity, the highest achieved distance was around 0.06 m for the temperature of 0°C. This distance decreases for higher temperatures,

dropping to around 0.05 m at 0°C. This does not align with our expectations based on the results from [12]. One possible explanation could be attributed to the differences in antenna topology. [12] employs optimized antennas compared to the omnidirectional ones used by our simulations. Hence, this needs to be investigated further. Therefore, for a 1 MHz-wide channel, wider distances between the nodes can be achieved for the lower salinity of 0.01 ppt (representing freshwater) than for other salinity values.



(a) Channel width 2 MHz, Salinity 0.01 ppt



(b) Channel width 2 MHz, Salinity 0.5 ppt

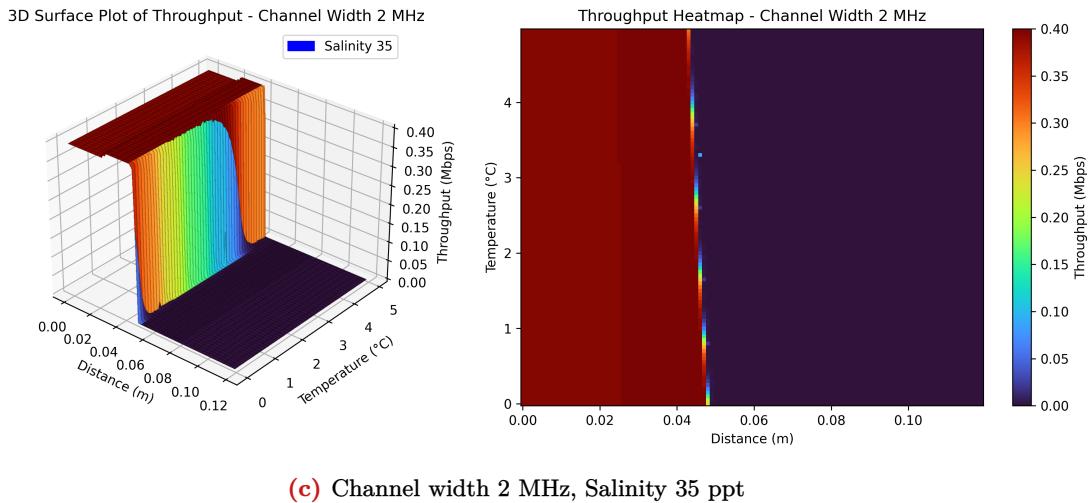


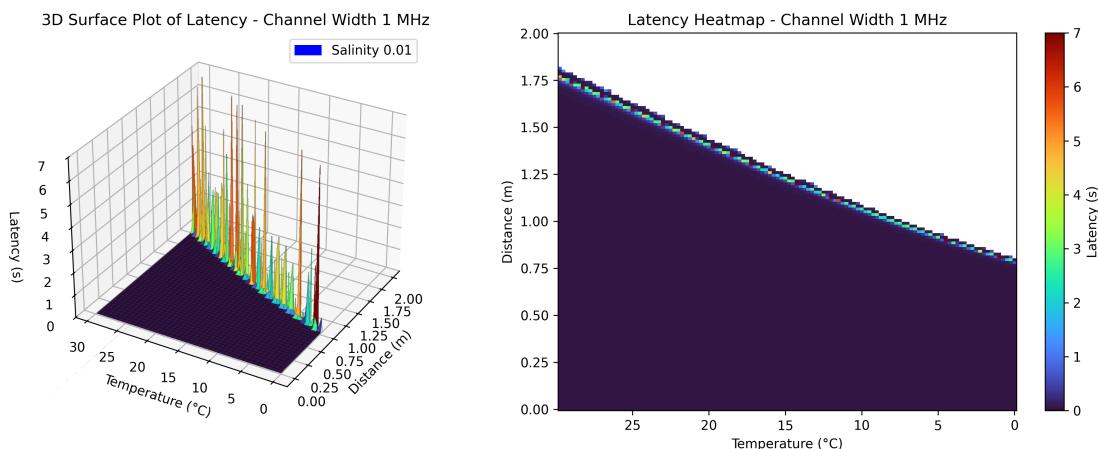
Figure 6.2: Mean throughput STA to AP vs. distance and temperature at 2 MHz in the peer-to-peer scenario

The 2 MHz-wide channels showed higher throughput than the 1 MHz-wide channels, reaching around 0.38 Mbps for all three different salinities.

For salinity 0.01 ppt (see Figure 6.2a), a significant drop can be seen for the distance between the nodes of around 0.6 m for 0 °C. Similarly, as for 1 MHz-wide channels, the distance increases with temperature, and at 30 °C, it reaches around 1 m. For salinity 0.5 ppt (Figure 6.2b), the drop is seen at a distance of less than 0.5 m. The distance is also slightly increasing with the increase in temperature, reaching a bit more than 0.5 m. This change is not as noticeable as in other cases. For salinity of 35 ppt (in Figure 6.2c), the highest distance of around 0.05 m was achieved for temperature around 0 °C, and it decreases with the increase in temperature.

In general, the throughput can be seen as higher for a 2 MHz-wide channel, but longer distances with non-zero throughput can be achieved for 1 MHz-wide channels - up to 1.75 m in 30 °C. Seawater is a poor environment for sensor deployment, since reliable links require very short node distances.

6.1.2 | Latency



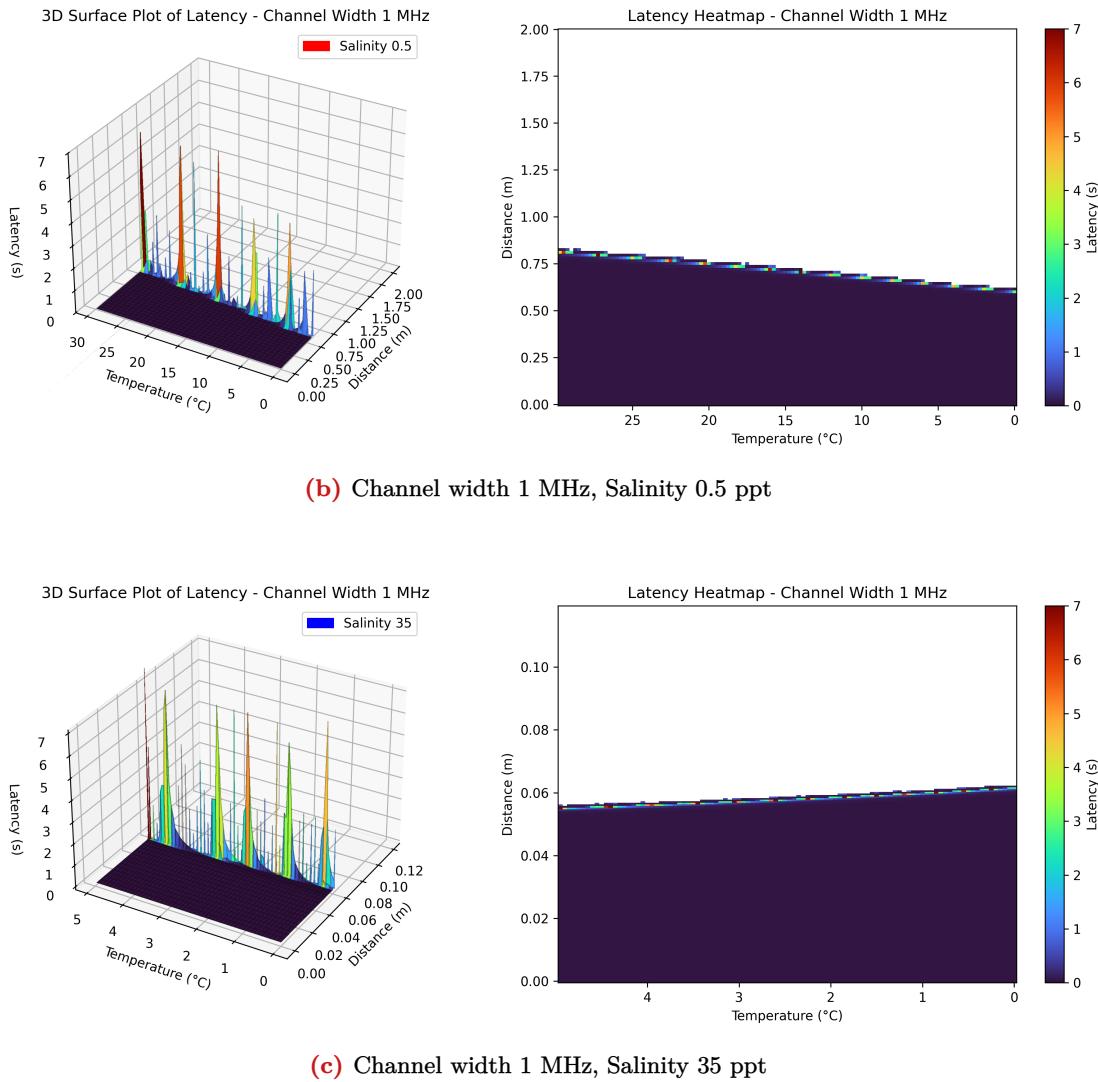
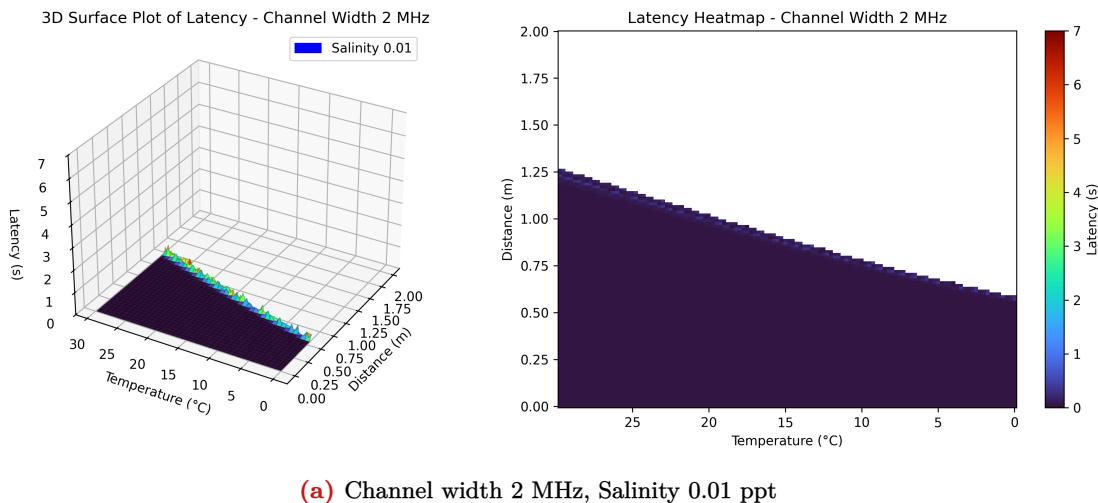


Figure 6.3: Mean latency STA to AP vs. distance and temperature for 1 MHz channel in the peer-to-peer scenario



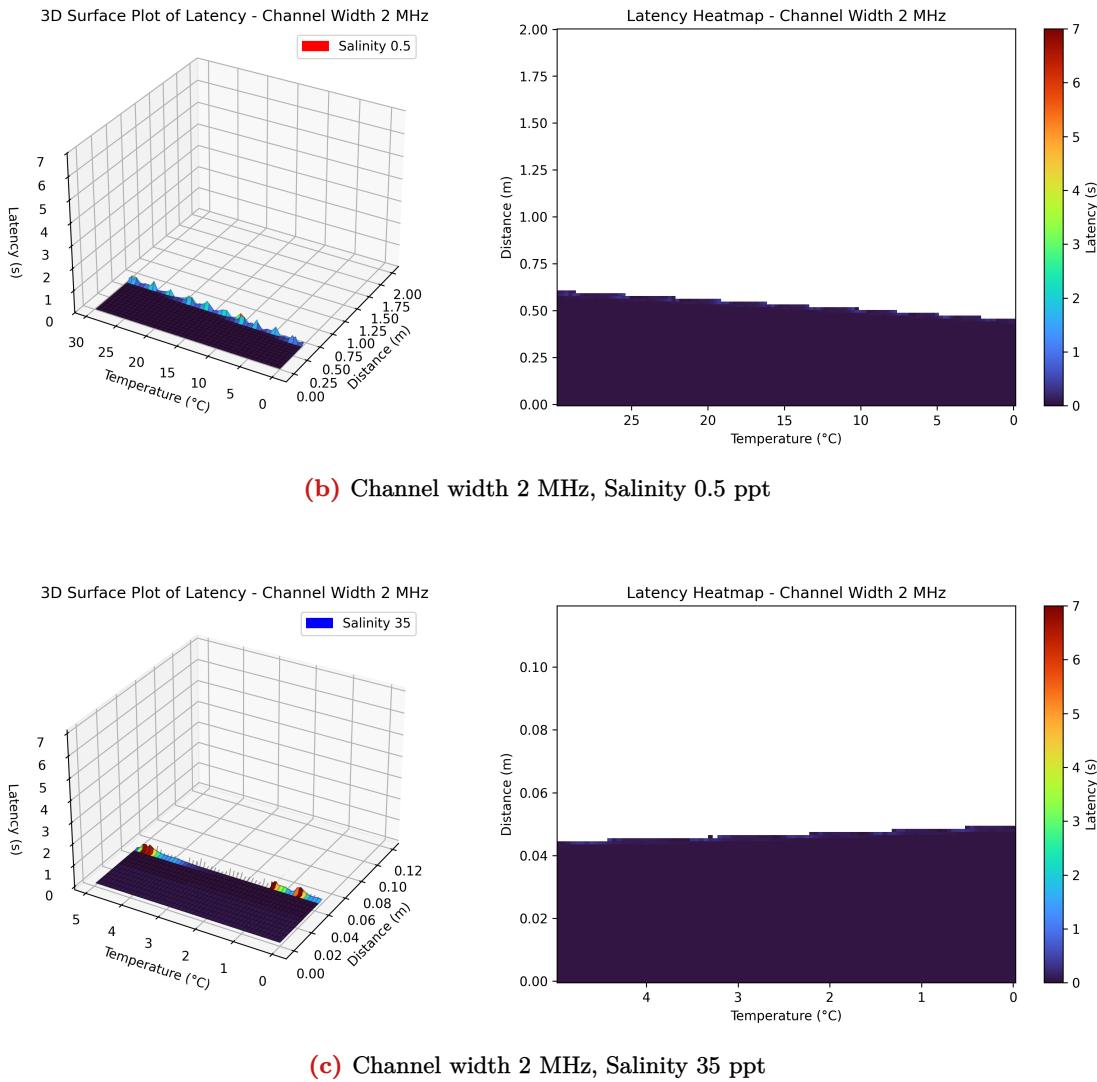


Figure 6.4: Mean latency STA to AP vs. distance and temperature at 2 MHz in the peer-to-peer scenario

Latency and throughput are inversely correlated: when throughput peaks, latency falls to 0 s; beyond the link boundary, latency spikes as throughput drops. This behavior is caused by increased distance between sensors and changes in temperature, which begin to degrade communication quality. As a result, packet retransmissions become more frequent due to unsuccessful transmissions. The blank regions in the latency heatmaps denote link failure (infinite latency). Therefore, as the communication link approaches its operational boundary, latency rises significantly, while throughput drops. This pattern is consistent across all scenarios: for all salinities and both channel widths (1 MHz and 2 MHz). In every case, latency exhibits an inverse relationship to throughput, which is expected.

Although the simulations for these two channel widths were run with the same precision, the increase in the latency for the 2 MHz-wide one is noticeably smaller than for the 1 MHz-wide channel. This is caused by the fact that a 1 MHz-wide channel is more robust at the link edge, tolerating weaker signals better than the 2 MHz channel. Hence, to obtain a higher increase in the latency on this boundary, the simulation for the 2 MHz-wide channel should be run with even more precision.

Notably, at certain boundary conditions, latency briefly lowers to standard before link failure, and anomalous behavior we defer to future work for further investigation.

6.1.3 | Power usage

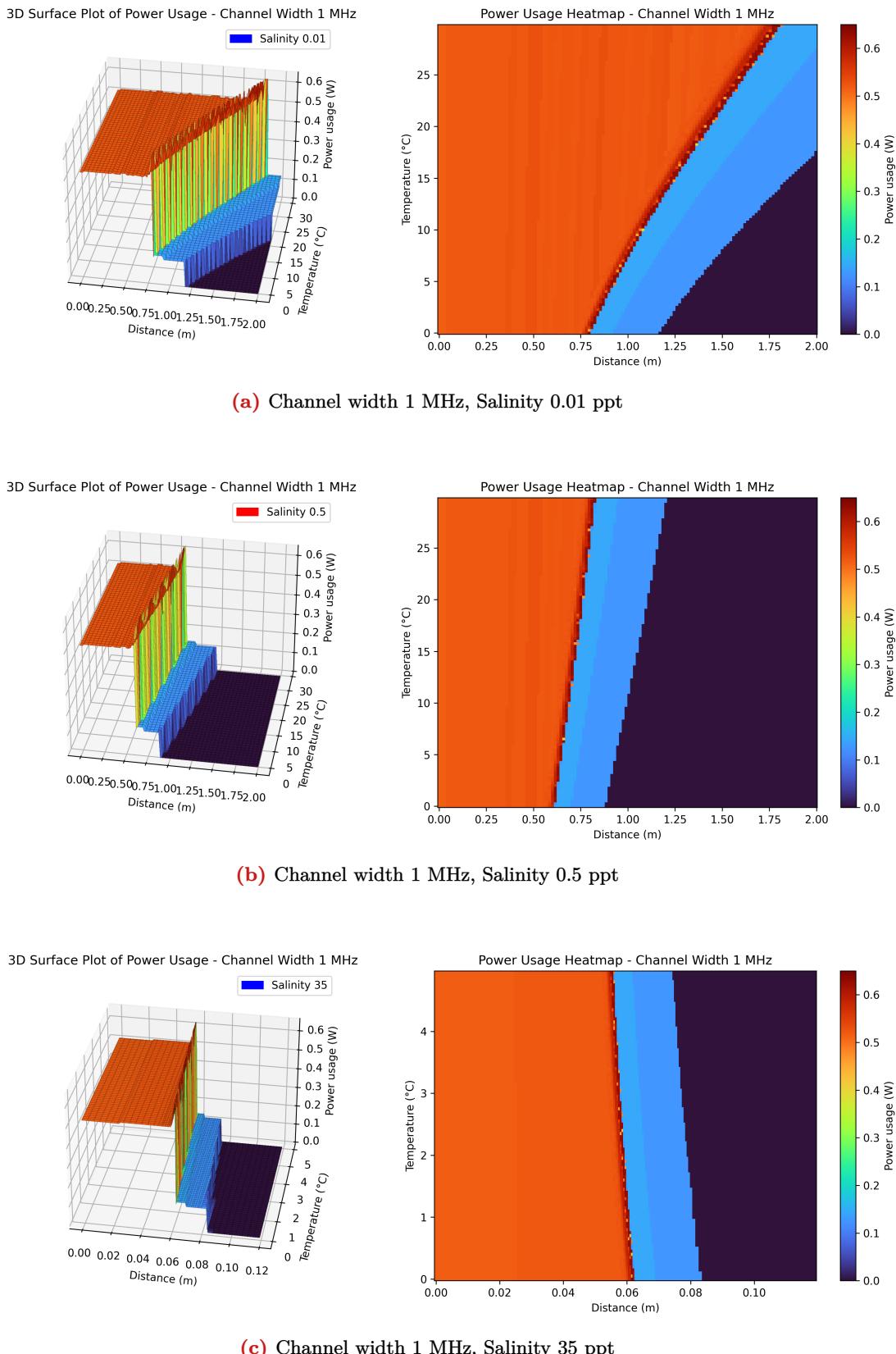
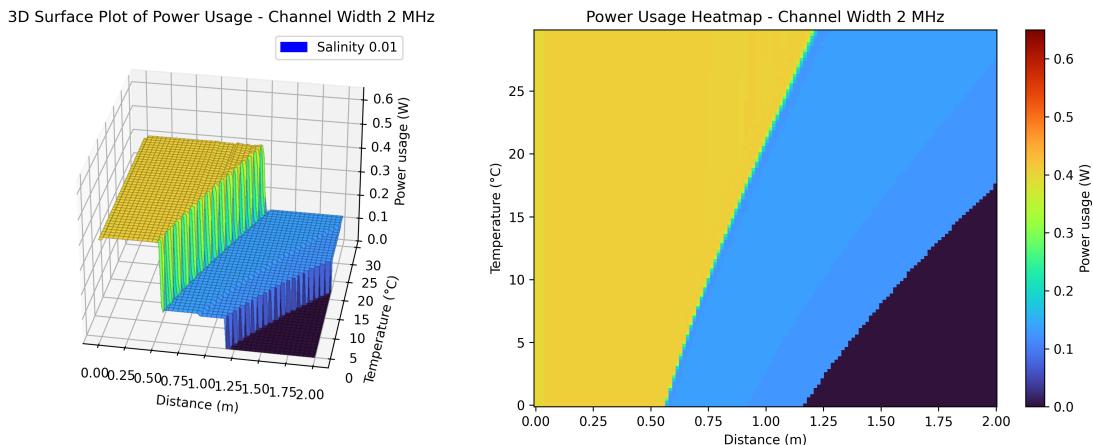


Figure 6.5: Mean power usage STA to AP vs. distance and temperature for 1 MHz channel in the peer-to-peer scenario

In the power usage graphs, the same trend can be visible, as in the previous metrics. We can see the highest power usage when the temperature and distance between the nodes is the most adequate for the data exchange (when the throughput is the highest and the latency is close to 0 s). In this case, the received signal exceeds the demodulation threshold, so frames are delivered normally, and energy use is the highest as full power is used for transmitting and receiving the packets. For the boundary of the reliable communication, we can see an increase in power, which also corresponds to the increase in latency. This is because the sensors retransmit the data more often, trying to maintain the communication in poor conditions. The signal still sometimes exceeds the demodulation sensitivity, and the packets can arrive.

When the signal gets weaker (throughput is 0 Mbps, so no packets arrive successfully), we can first notice a blue plateau in all of the graphs. This is because the signal has dropped below the demodulation sensitivity but remains above the Energy Detection Threshold (CCA1), (which initial value is set to -96dBm in **YansWifiPhy** [2]) - no packets ever decode (throughput remains at 0 Mbps), but still, it is strong enough to detect the packet. This causes the receiver to briefly stay active, consuming power. Meanwhile, the sender's MAC layer, not receiving any acknowledgment (ACK), assumes the packet was lost and begins a retry cycle: it waits (backoff), retransmits the packet, waits again for an ACK, and then times out. This loop repeats until the retry limit is reached. Throughout this process, both devices remain in active transmit and receive states, consuming significant energy, even though no data is successfully exchanged. What can also be interesting to notice is that the blue plateau is slightly decreasing with distance between the nodes. This is because the signal is weaker and only exceeds the modulation sensitivity (CCA2) threshold, which only allows for preamble detection on the AP.

When the power usage drops to 0 W, the signal even drops below the modulation sensitivity threshold, so the sensors do not wake up to listen and they immediately return to the IDLE or SLEEP states. This behavior can be observed in all of the graphs.



(a) Channel width 2 MHz, Salinity 0.01 ppt

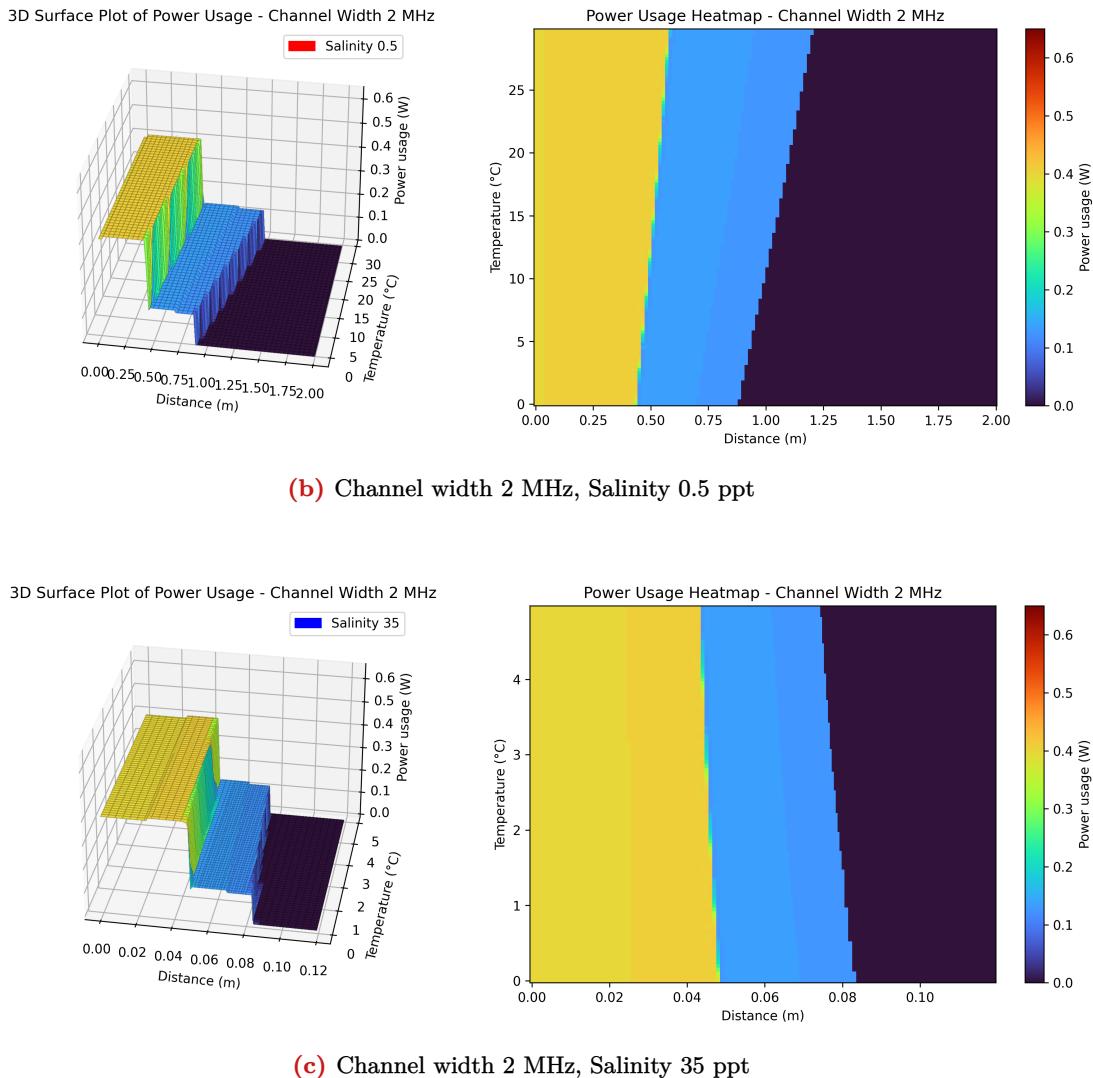


Figure 6.6: Mean power usage STA to AP vs. distance and temperature at 2 MHz in the peer-to-peer scenario

The 1 MHz channels during reliable transmission use the power of 0.5 W on average for all salinities (see Figure 6.5. An intermediate plateau at approximately 0.1 W occurs when the signal is below the demodulation threshold but above the carrier-sense threshold, before falling to 0 W. For the 2 MHz channels, this power usage is lower, with the average of about 0.37 W for all salinities (see Figure 6.6. The intermediate plateau reaches around 0.1 W as well and slightly decreases before reaching 0 W.

6.1.4 | Power efficiency

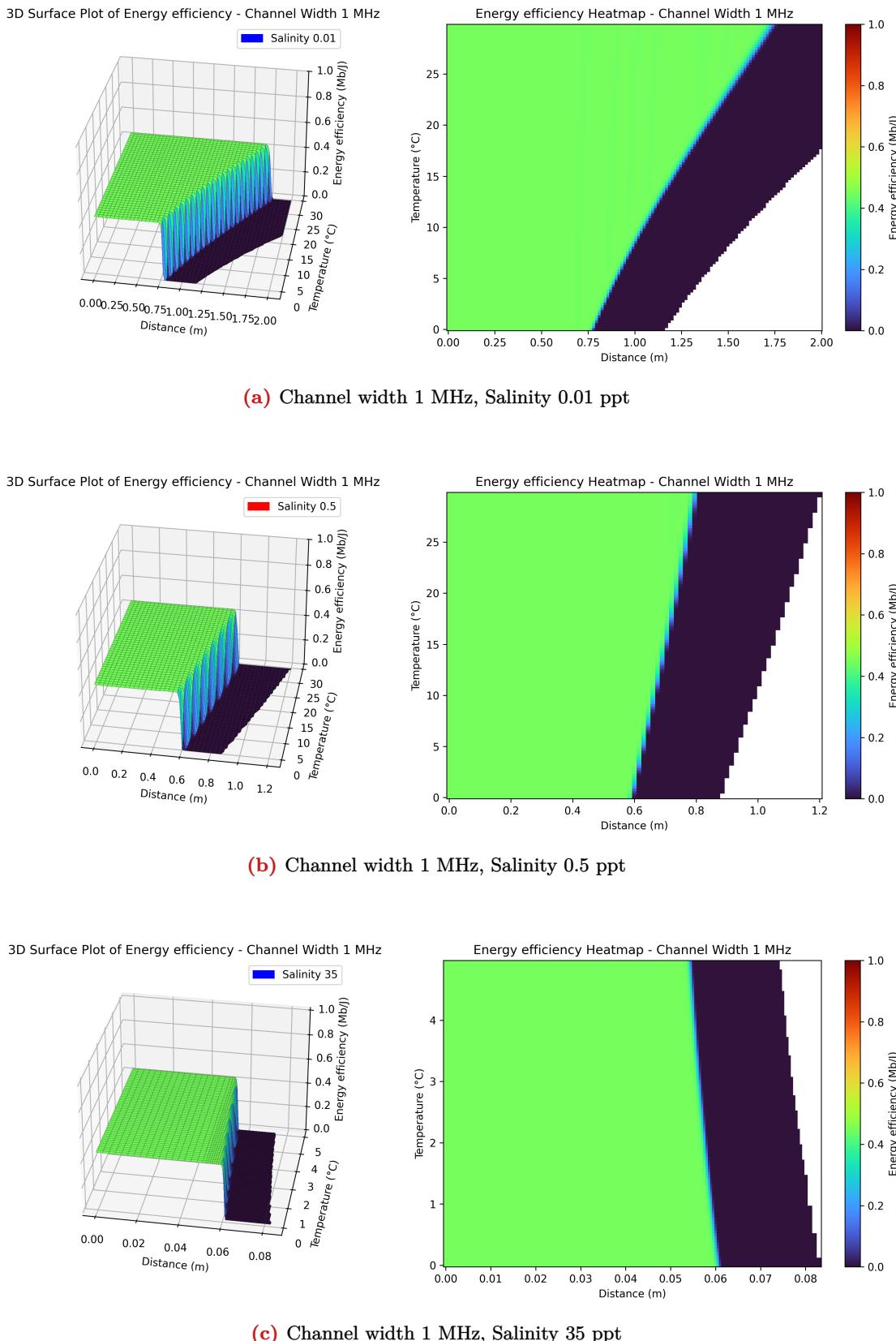
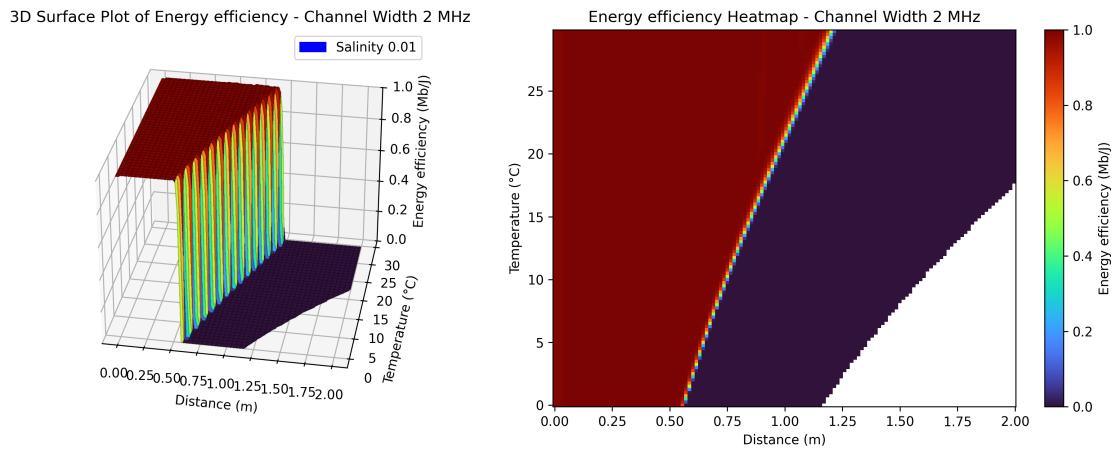
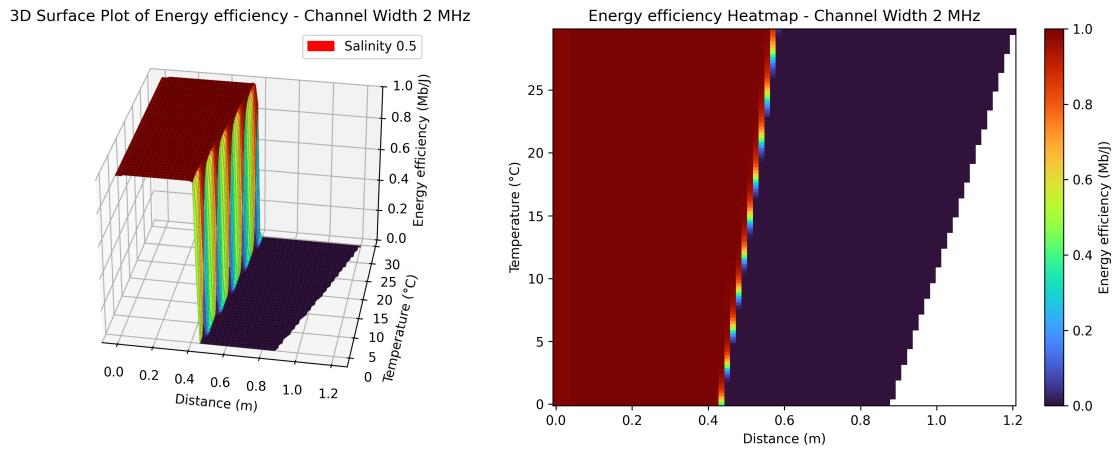


Figure 6.7: Mean energy efficiency STA to AP vs. distance and temperature for 1 MHz channel in the peer-to-peer scenario

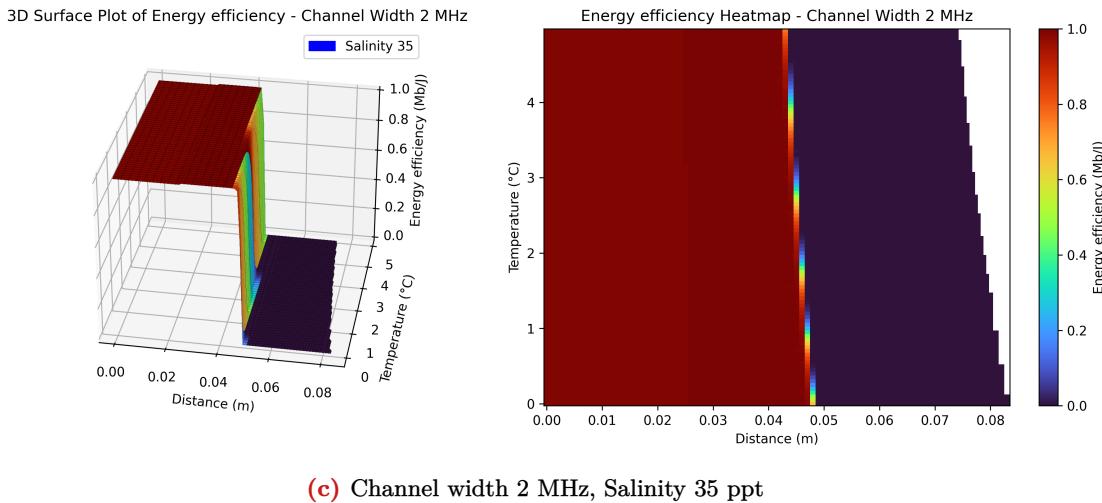
In terms of energy efficiency, 1 MHz-wide channels achieve the energy efficiency on average of around 0.4 Mb/J for the simulations with the temperature and distance supporting reliable communication. When the communication is no longer possible (and throughput is 0 Mbps), the energy efficiency drops down to 0 Mb/J as expected.



(a) Channel width 2 MHz, Salinity 0.01 ppt



(b) Channel width 2 MHz, Salinity 0.5 ppt



(c) Channel width 2 MHz, Salinity 35 ppt

Figure 6.8: Mean energy efficiency STA to AP vs. distance and temperature at 2 MHz in the peer-to-peer scenario

Similar behavior can be observed for 2 MHz-wide channels, however, they seem to be more energy efficient, as the energy efficiency reaches around 0.95 Mb/J when the throughput is the highest. This behavior is expected as in this situation the throughput is higher and the power usage is lower than the one observed for a 1 MHz-wide channel. It can be observed again that different salinities do not affect these values, as they only limit the communication range. For shorter distances with the need for lower energy efficiency, 2 MHz-wide channels are a better option.

6.1.5 | Conclusions

Overall, it can be concluded that the 2 MHz-wide channels are a better choice for short-range use cases with the need for high throughput, up to around 0.75 m with salinity 0.5 ppt and temperature around 30 °C, and up to 1.5 m in salinity of 0.01 ppt and the same temperature. They also perform better in terms of energy efficiency. On the other hand, 1 MHz wide channels are a better choice for longer distance communication, at most 1.75 m for a temperature of 30 °C, salinity 0.01 ppt, and a distance of around 0.6 m for a salinity of 0.5 ppt and the same temperature. For this case, the throughput achieves 0.2 Mbps.

Seawater imposes severe range limitations on underwater EM communication, restricting reliable links to just a few centimeters. Consequently, practical applications are confined to near-contact, waterproof operations: for instance, an AUV docking system with a submerged data-harvesting station to transfer information without exposing electrical contacts to corrosive saltwater.

6.2 | Sensors-to-AP

We advise the reader to disregard the results where the sensor count is larger than 64 because in those cases, a single TIM group consists of more than 64 stations, which goes against the specification of the protocol [46].

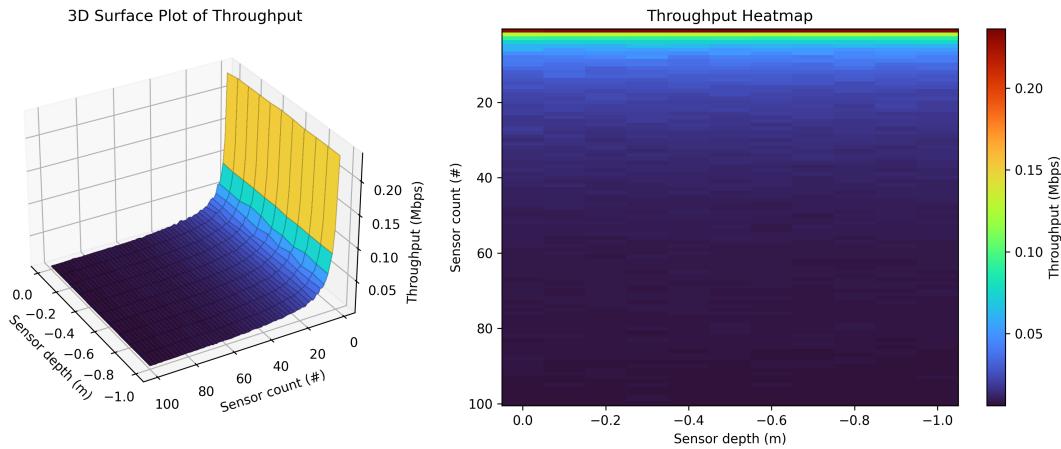


Figure 6.9: Mean throughput STA to AP vs. sensor depth and sensor count in the Sensors-to-AP scenario

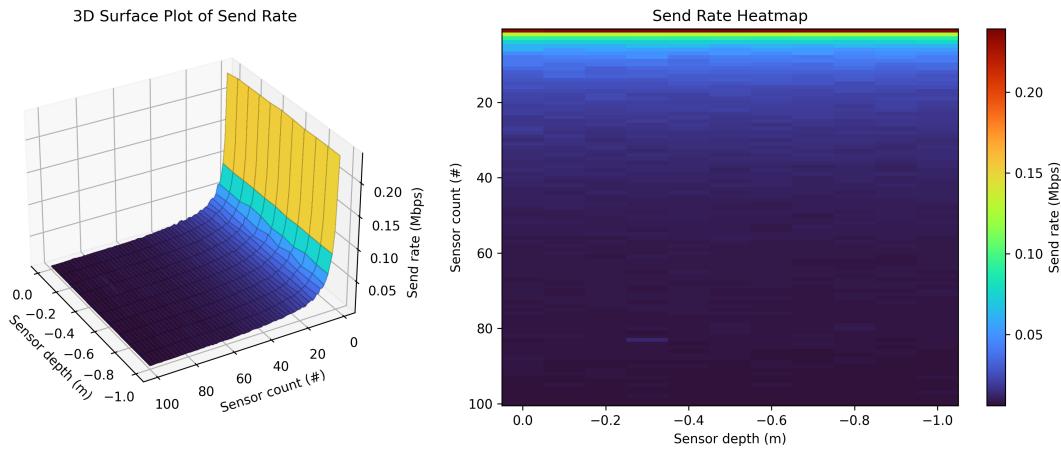


Figure 6.10: Mean send rate STA to AP vs. sensor depth and sensor count in the Sensors-to-AP scenario

With a single-slot RAW group, adding more sensors increases channel occupancy, resulting in longer queuing delays and causing TCP to reduce the sending rate (Figure 6.10), which reduces the per-station throughput (Figure 6.9). The distance from the AP to the stations (if all sensors are in range and there is no hidden node problem) does not significantly impact the mean throughput.

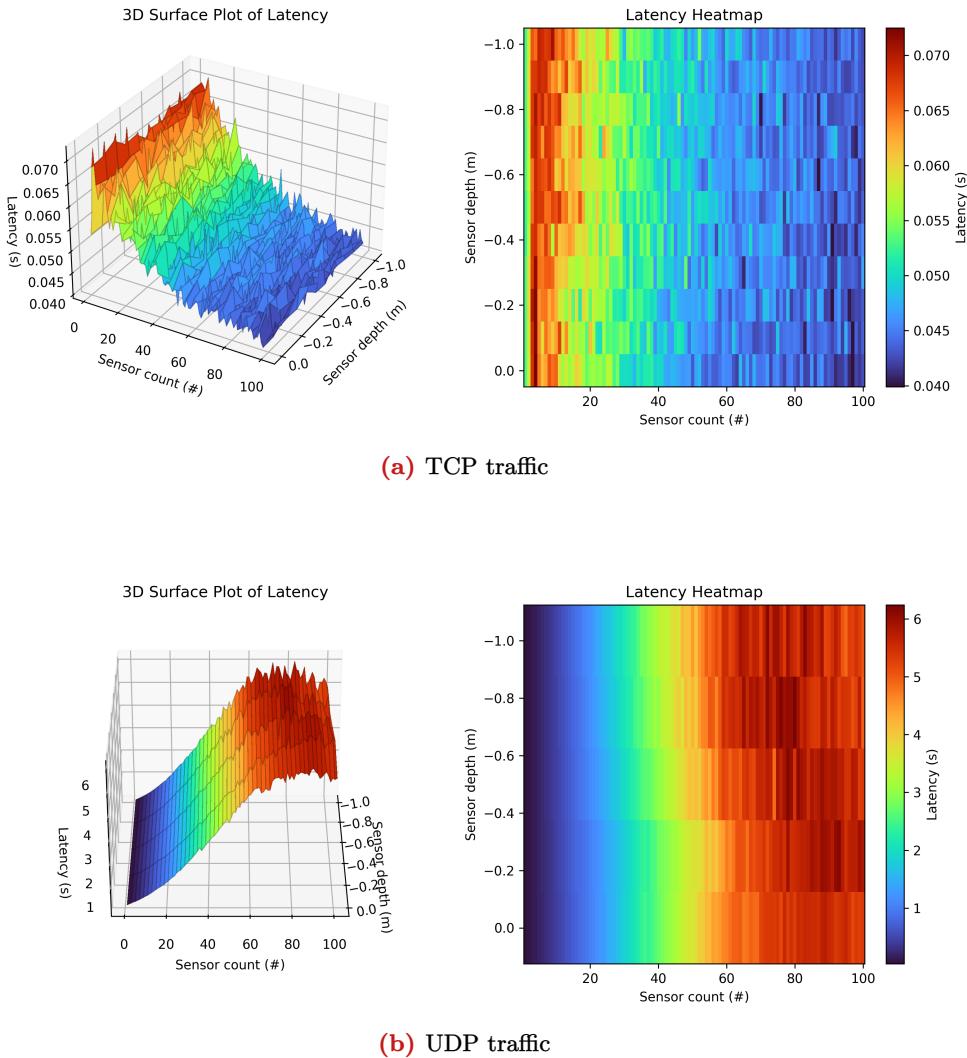


Figure 6.11: Mean latency STA to AP vs. sensor depth and sensor count in the Sensors-to-AP scenario

We define latency as the end-to-end delay of a successfully delivered transport layer packet, without accounting for retransmissions, which can occur if TCP is used as the transport layer protocol.

Figure 6.11a presents unexpected results. One would expect the mean latency to increase as the number of sensors increases because there would be more MAC-level queuing delays at the stations due to the busy channel. We hypothesized that TCP's congestion control reduces the send rate under load, reducing queueing delays and improving latency. To test this, we repeated the simulation using UDP, which showed the expected latency increase with more sensors (Figure 6.11b). Hence, the most plausible cause of the latency decrease is the TCP congestion control algorithm suppressing the sending rate as it detects congestion. Again, sensor depth (assuming all sensors are in range and no hidden node effects apply) does not significantly impact the mean latency.

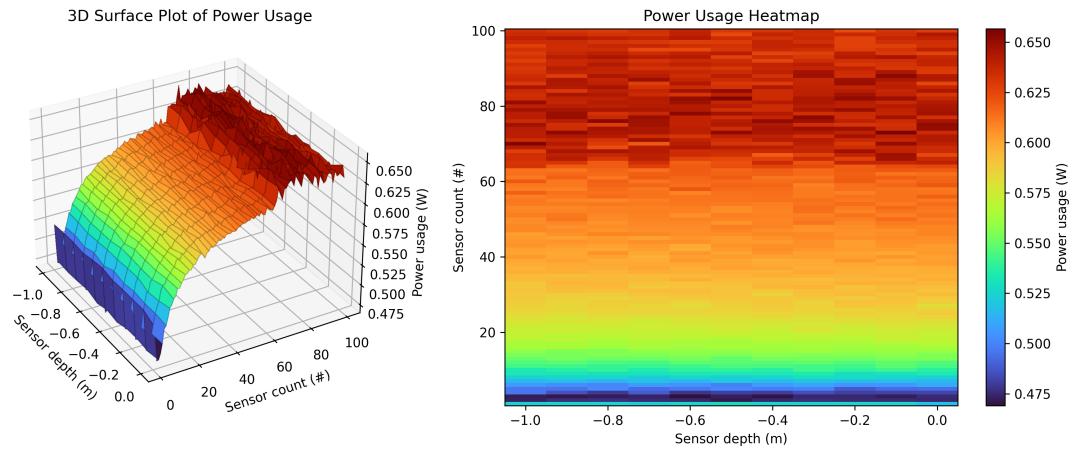
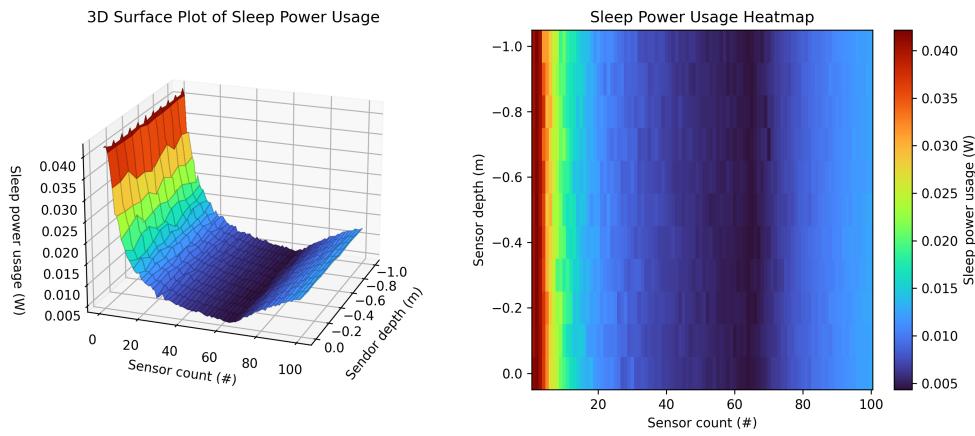
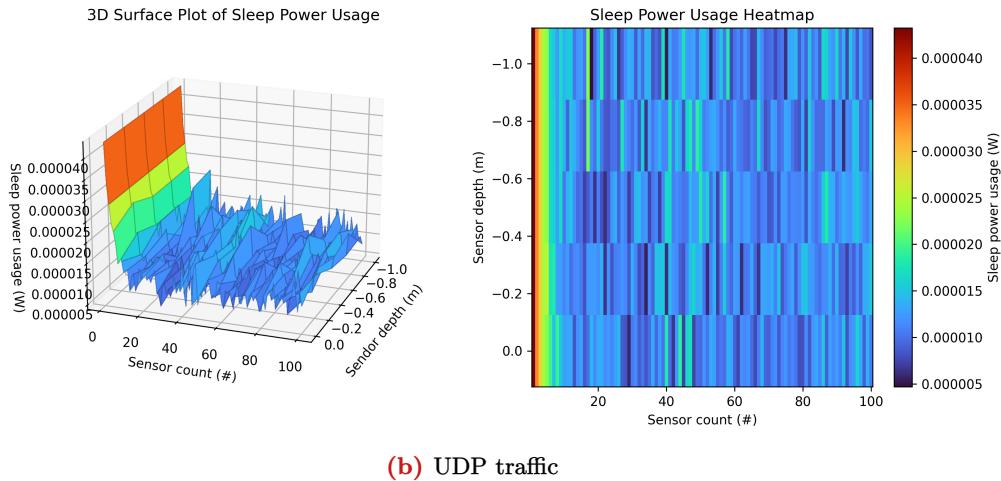


Figure 6.12: Mean power usage STA to AP vs. sensor depth and sensor count in the Sensors-to-AP scenario

We can measure the energy consumption for each category of a station's operating states (**SLEEP**, **IDLE**, **TX**, **RX**, **CCA_BUSY**, and **SWITCHING**) by tracking the amount of time spent in each state along with the energy usage in between two state transitions. In our experiments, **SWITCHING** energy is always zero, since we do not perform channel hopping as HaLow lacks multi-channel support due to global spectrum restrictions.

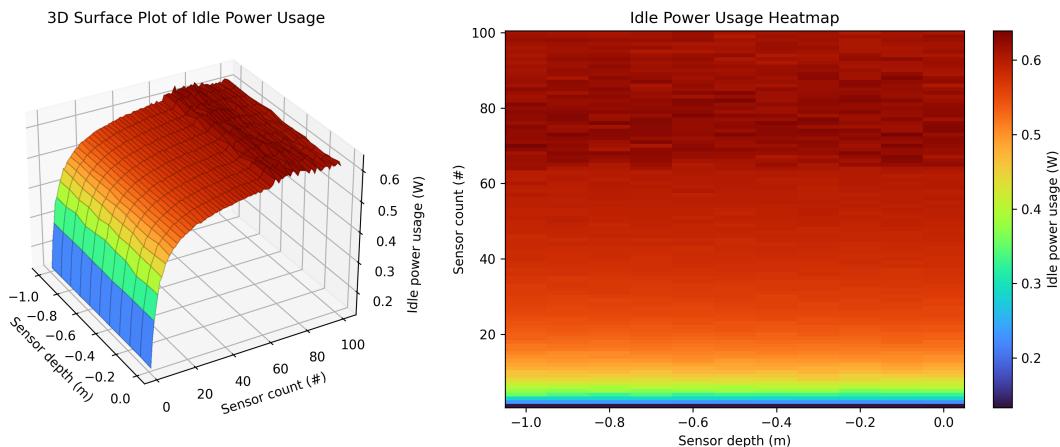




(b) UDP traffic

Figure 6.13: Mean SLEEP-state power usage consumption per sensor

The power usage in the SLEEP state decreases with the increase of the number of sensors (Figure 6.13a). This can be explained by the fact that TCP suppresses the sending rate as the number of stations increases, which leads to some stations not having packets to transmit. Therefore, the sensors transition to SLEEP. We can verify it by looking at the power usage graph in the SLEEP-state for UDP (Figure 6.13b), where stations always have something to transmit, and observing that the power usage remains fairly constant as the count of sensors increases. However, we do not have an explanation for why power usage in SLEEP-state for UDP for very few sensors is high compared to other configurations. This needs to be investigated further.

**Figure 6.14:** Mean IDLE-state energy power usage per station (TCP traffic)

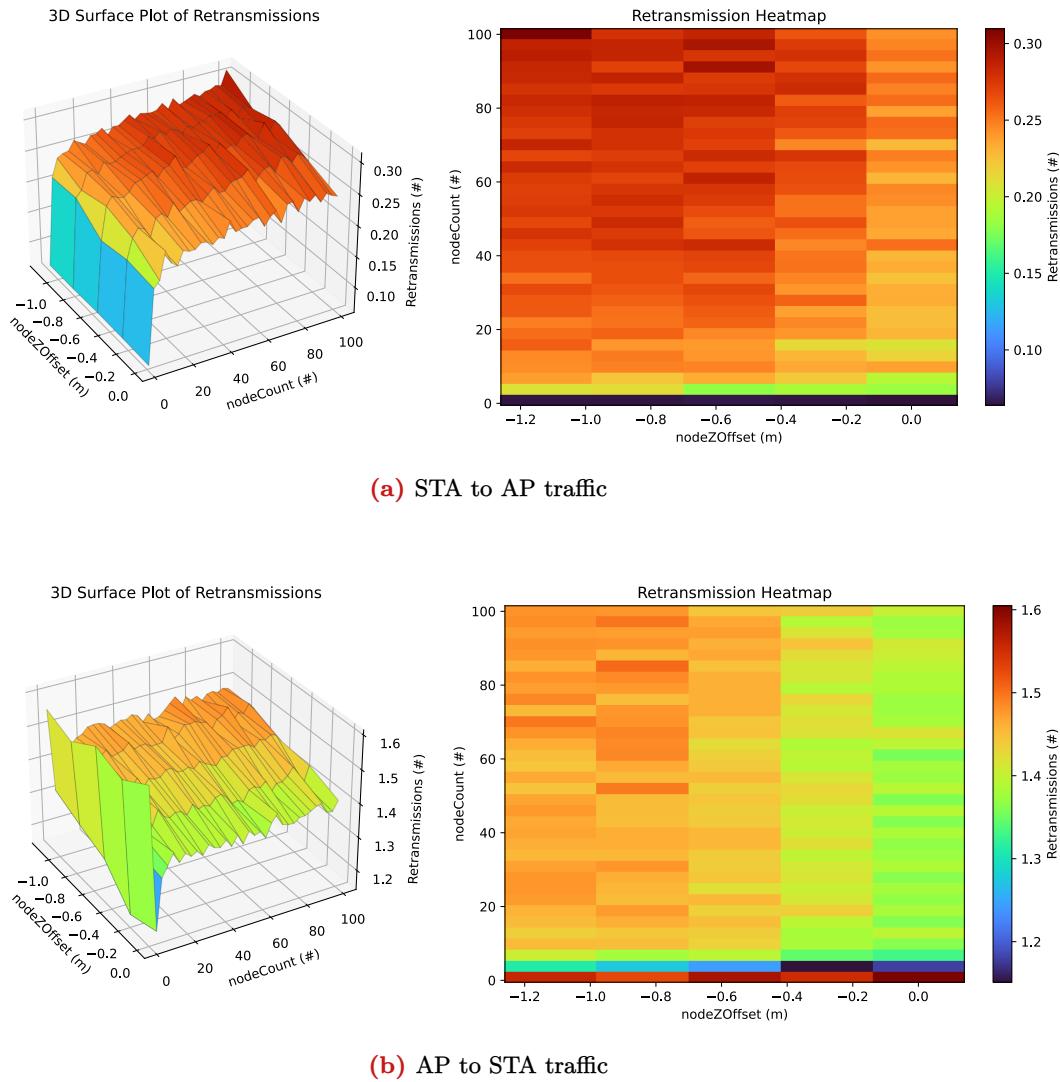


Figure 6.15: Mean number of retransmissions per sensor

IDLE energy per sensor can increase for two reasons:

1. After the sensor has started transmitting, there is a collision, which makes it pick a larger back-off period.
2. It waits longer for the TCP acknowledgments.

To see the reason, let's analyze the retransmissions. We can see that STA to AP retransmissions increase as the number of stations increases (Figure 6.15a), whereas this pattern is not present in AP to STA traffic (Figure 6.15b), which consists solely of acknowledgments and remains constant with the increase of sensor count. Hence, the only explanation for higher IDLE energy with the increase of sensor count is the collisions after the transmissions start, which causes stations to pick a larger back-off period and, hence, stay in IDLE state for longer.

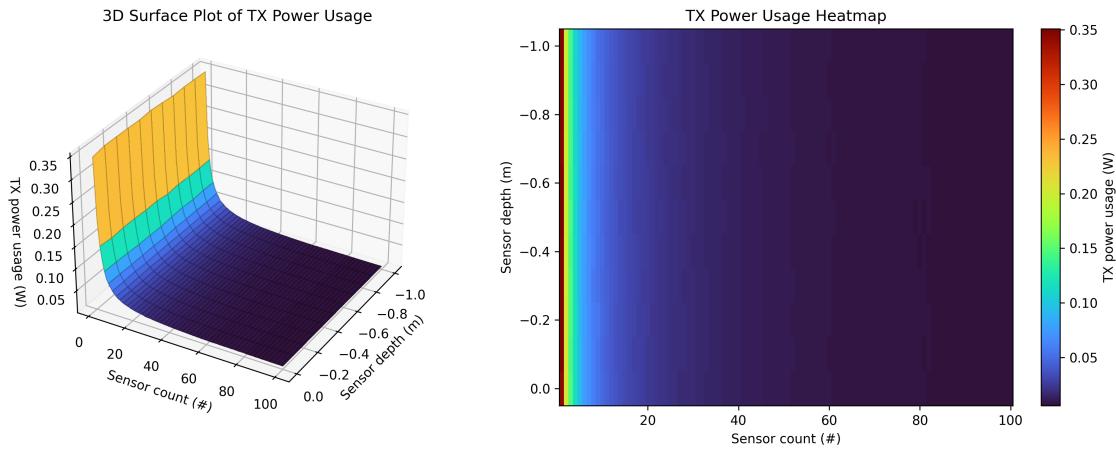


Figure 6.16: Mean TX-state power usage per station (TCP traffic)

The power usage pattern in TX-state (Figure 6.16) follows the pattern in the send rate graph, which is as expected.

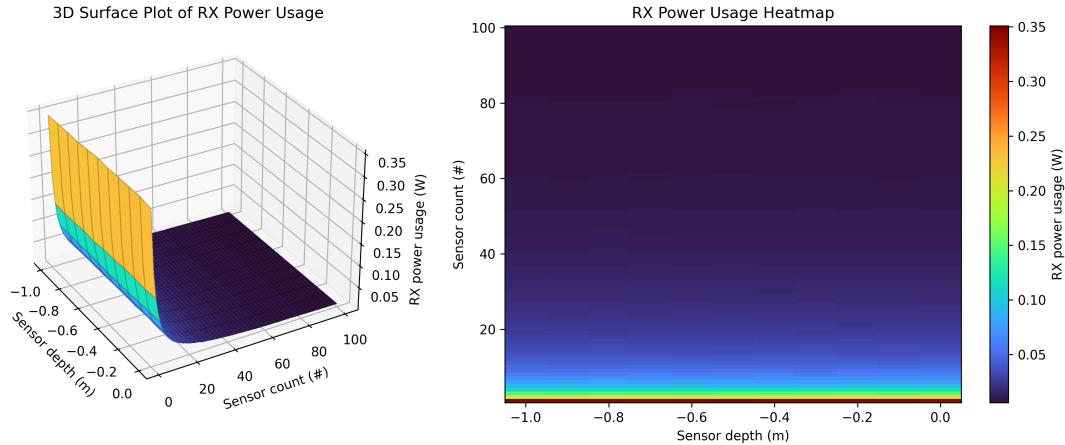


Figure 6.17: Mean RX-state power usage per station (TCP traffic)

The increase in power usage in RX-state with the increase in sensor count can be explained by receiving packets that are transmitted by other stations. Since we didn't log PHY-level packets between stations, we cannot verify if the explanation is correct, so we leave it as future work.

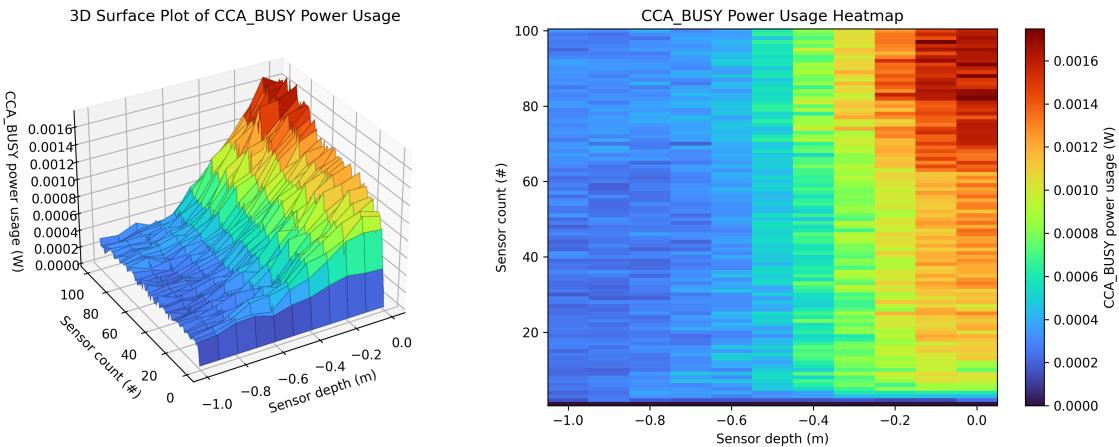


Figure 6.18: Mean CCA_BUSY-state power usage per station (TCP traffic)

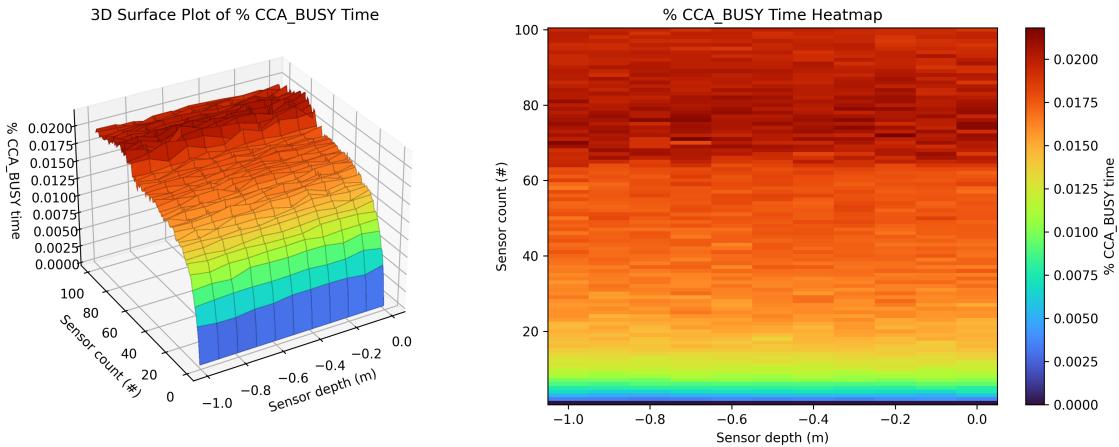


Figure 6.19: Mean percentage of time CCA_BUSY is active relative to simulation time

The graph for power usage in CCA_BUSY-state (Figure 6.18) shows an unusual pattern, showing that STAs use more power as the AP moves closer to them. Even more intriguing is that the time spent in CCA_BUSY per station (Figure 6.19) remains constant as the sensor depth varies. One possible explanation could be that CCA_BUSY spends more energy when the station is receiving a stronger signal compared to when it receives a weaker signal that is still above the CCA threshold. Then, when the AP is closer and the sensors are at a smaller depth, the stations receive a stronger signal compared to when the AP is farther away. As we didn't verify this hypothesis, further investigation is required.

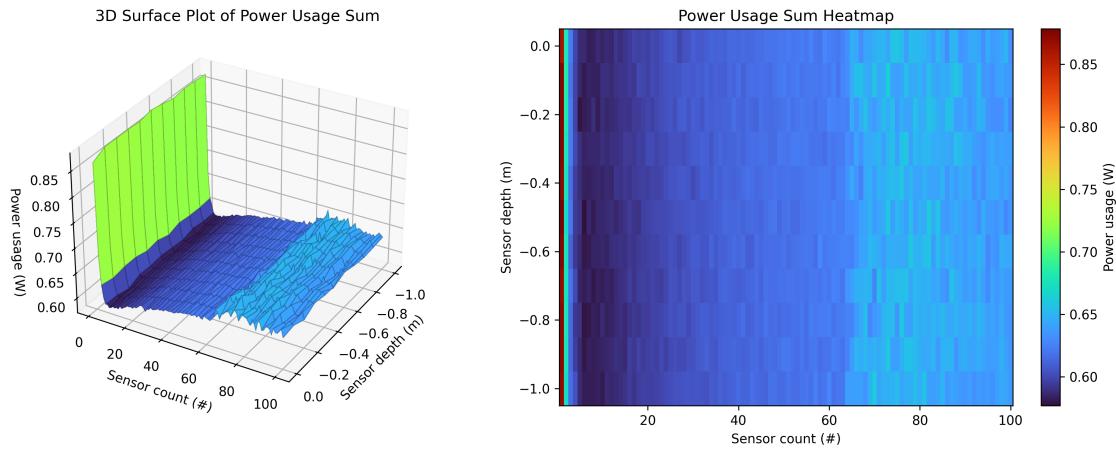


Figure 6.20: Sum of power usage graphs over all states

The sum of power usage graphs over all states (Figure 6.20) does not match the total reported power usage (Figure 6.12) for the reasons described in 4.2. This discrepancy requires further investigation.

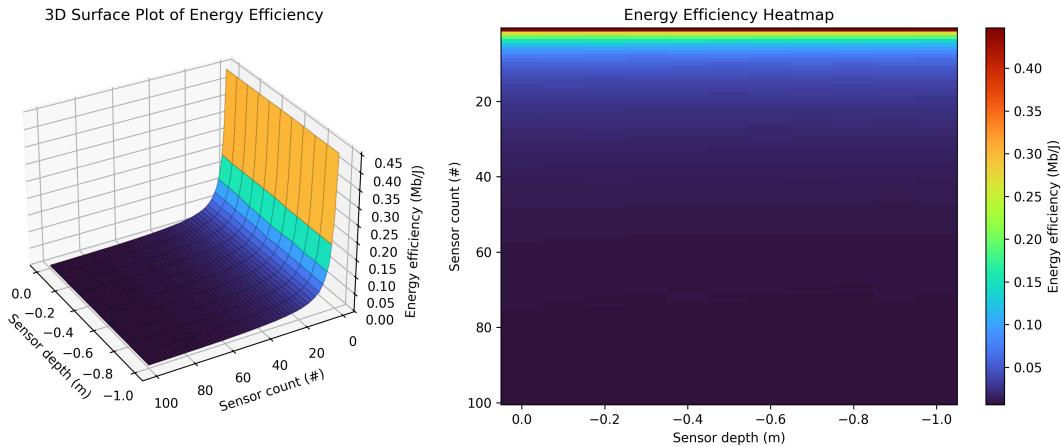


Figure 6.21: Mean energy efficiency STA to AP vs. sensor depth and sensor count in the Sensors-to-AP scenario

The mean energy efficiency graph (??) can be explained by looking at the patterns in the throughput (??) and power usage graphs (Figure 6.12). As the number of sensors increases, the throughput per sensor goes down, but the power usage per sensor increases. Hence, the energy efficiency must decrease, which can be seen in the graph.

6.3 | Data mulling

6.3.1 | Hidden node problem

This is the only scenario of the three for which the hidden node problem can be observed, due to the arrangement of the sensors such that in some configurations (sensor spacing larger than 1.4 meters) more than one sensor is in range of the AUV (AP) but not within range of each other, leading to multiple sensors communicating at the same time and creating additional interference resulting in communication difficulties. To analyze this, a simple Python simulation was executed to identify the mean of the number of hidden nodes and the ratio of hidden nodes to connected nodes with respect to the sensor spacing and the AUV speed (resulting graphs can be seen in Figure Figure 6.22). These metrics were evaluated to

identify recurring patterns resulting from the hidden node problem that could emerge in the later results of the ns-3 simulations.

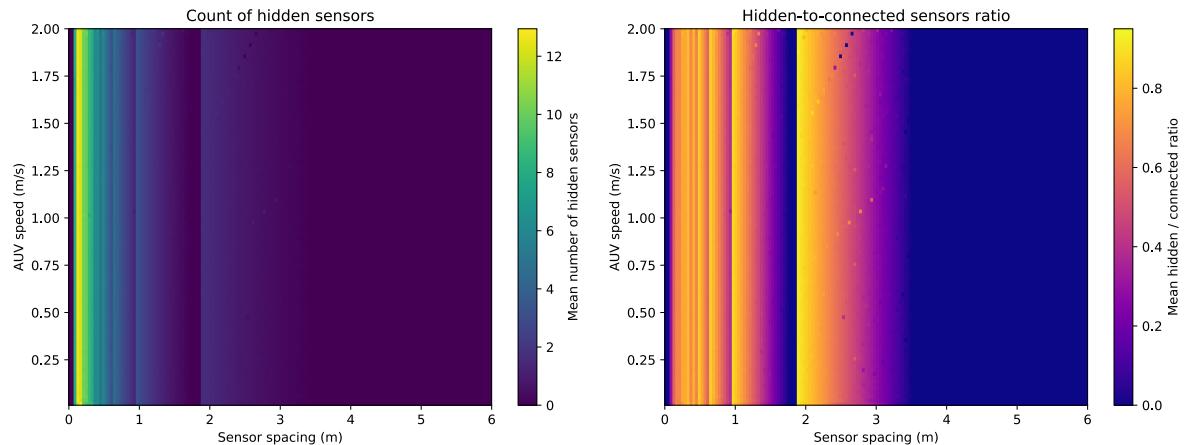
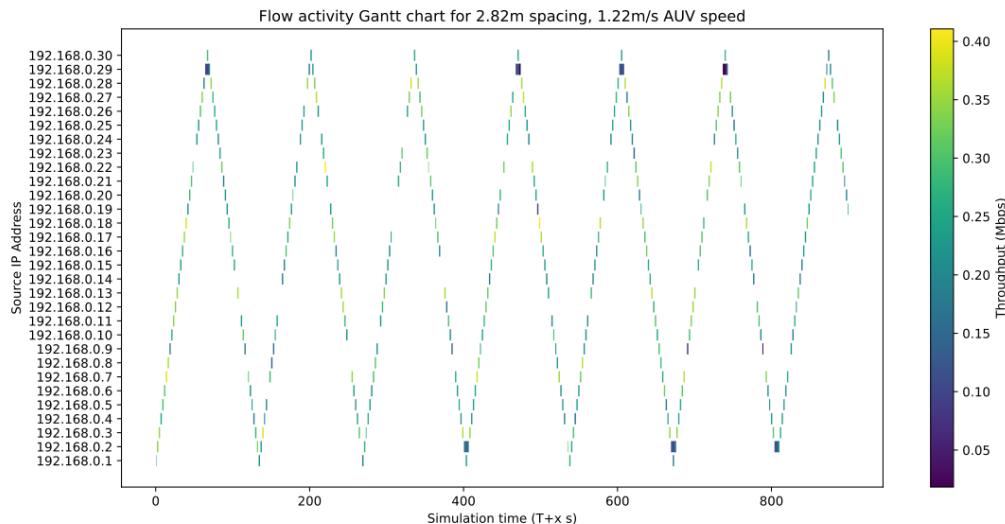


Figure 6.22: AUV hidden nodes results

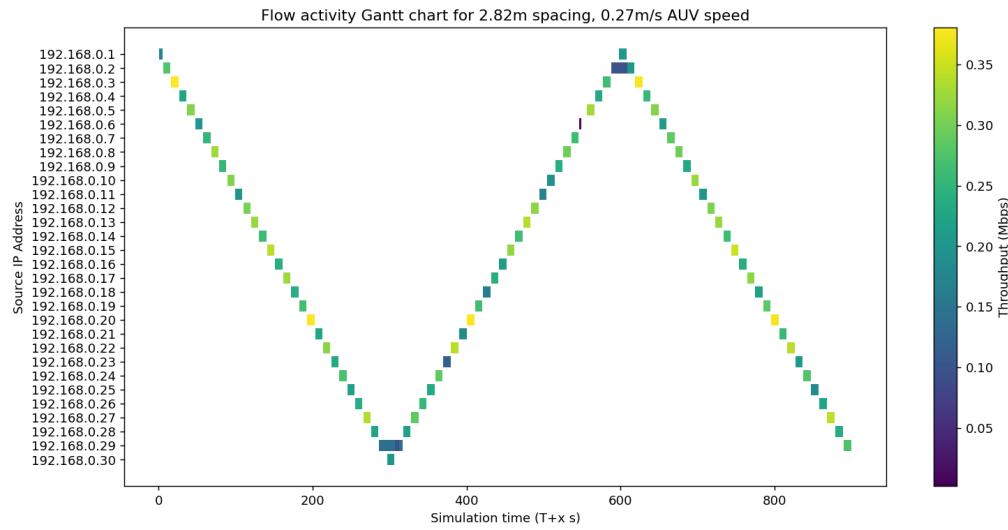
The simulation results, depicted in the graph, indicate two distinct patterns that should be noticed. The first pattern appears to be only dependent on the sensor spacing. A sudden rise in the ratio of hidden-to-connected nodes at sensor spacing of 1.9 m, and then gradually going to 0 at around 3.4 m. This pattern is repeated at 0.9 m to 1.8 m, at 0.65 m to 0.95 m, at 0.55 m to 0.63 m, etc.

The second pattern can be more easily observed in the gradients observed in the first pattern. This second pattern consists of data points on diagonal lines showing the interdependence of the hidden node count to both the sensor spacing and AUV speed. These 'diagonals' can be observed from (0.75 m/s, 0.95 m) to (1.25 m/s, 3.4 m), from (1.5 m/s, 0.95 m) to (2 m/s, 2.7 m), etc. Overall, these graphs identify recurring patterns caused by the hidden node problem, allowing differentiation between outcomes driven by this issue and those influenced by other phenomena.

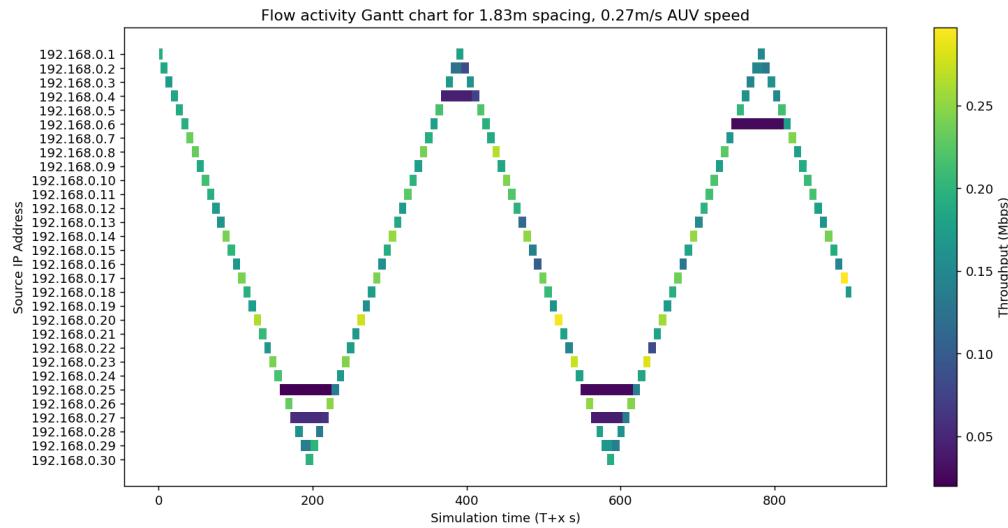
6.3.2 | Connection windows



(a) Connection timeline per node for sensor spacing of 2.82 m, AUV speed of 0.27 m/s



(b) Connection timeline per node for sensor spacing of 1.83 m, AUV speed of 1.22 m/s



(c) Connection timeline per node for sensor spacing of 1.83 m, AUV speed of 0.27 m/s

Figure 6.23: Connection windows comparison

For the connection window graphs three graphs are compared, sharing either sensor spacing or AUV speed with each other. Looking at the first pair (Figure 6.23b and Figure 6.23a), it can be seen that for higher speed we have more connections for shorter time with relatively similar throughput compared to the graph with slower AUV speed. Meanwhile, for the second pair, Figure 6.23c shows that the smaller sensor spacing (and the same speed) results in more connections with the same time per connection and lower mean throughput. An important part of this graph is the longer connection times, which can be observed for sensors with IPs 192.168.0.25 and 192.168.0.27 at 200 s and 600 s. After a deeper investigation, the results observed for these sensors were considered to be artifacts of the way the communication statistics are currently captured: **FlowMonitor** [2] does not close the connection for those sensors even after they are out of range with the AUV, hence maintaining the connection without transmitting any data and thus reducing the mean throughput, as can be seen by the purple lines in Figure 6.23c.

6.3.3 | Throughput

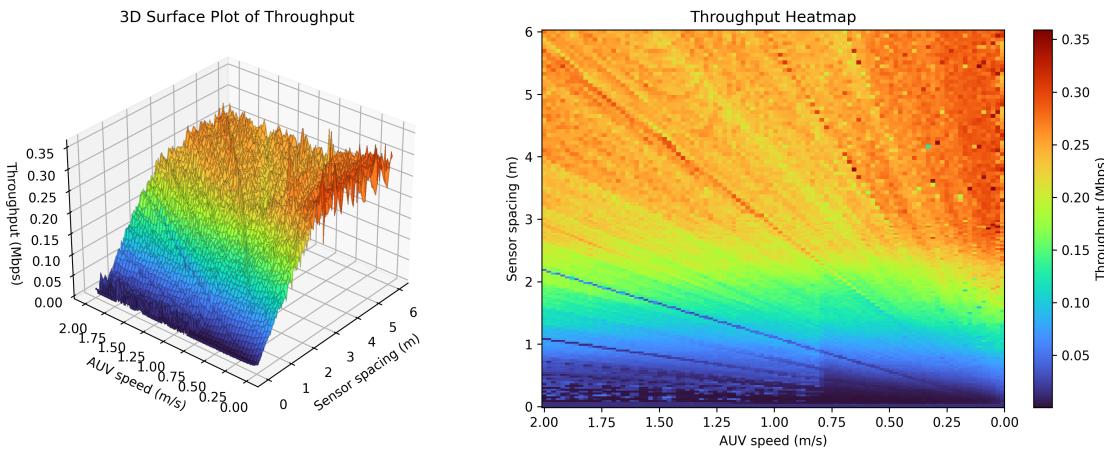


Figure 6.24: Mean throughput STA to AP vs. sensor spacing and AUV speed

For the data mulling scenario, the maximum throughput reached is between 0.3 to 0.35 Mbps in the region with both large sensor spacing (2.8 m to 6 m) and low AUV speed (0 m/s to 0.4 m/s). This result is expected as the range of HaLow at 20 °C and 1 MHz-wide channels is around 1.4 m (refer to Figure 6.1), thus at sensor spacing larger than 2.8 m the interference should be significantly lower. Moreover, a lower speed would allow more time for the sensors to associate with the AUV and transmit more data before they again go out of range. (Refer to Figure Figure 6.23)

It is also interesting to notice that generally for low spacing (0.1 to 0.3 meters) the throughput is close to zero; however, for higher speeds (0.8 to 2 m/s) the throughput is increased, reaching around 0.05 Mbps, which is not significant but interesting to notice. Finally, it is important to note that the 'diagonal' pattern of the Section 6.3.1 Hidden node problem is very prominent, meaning that it affects the throughput negatively.

6.3.4 | Latency

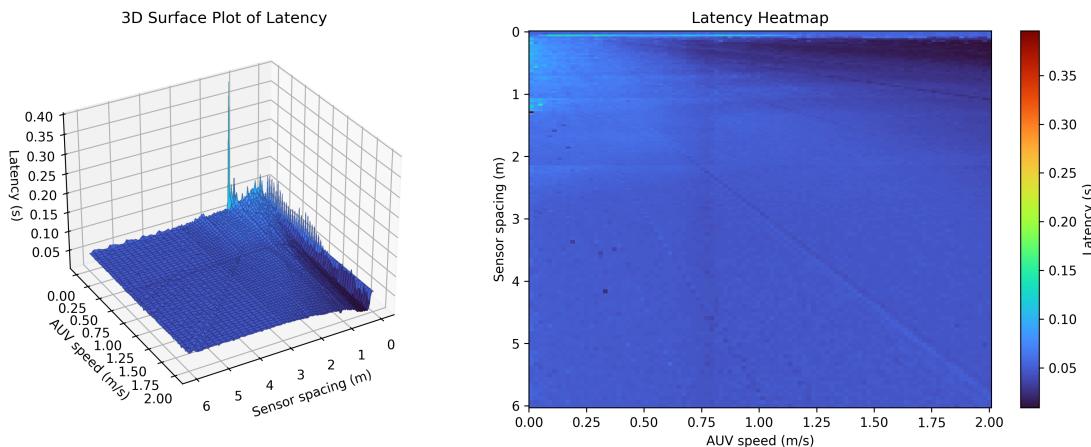


Figure 6.25: Mean latency STA to AP vs. sensor spacing and AUV speed

As the graph (Figure 6.25) depicts, the latency remains mostly constant at around 0.08 s for sensor spacing above 1.2 m, except for the diagonal and vertical patterns, which are consistent with the Section 6.3.1 Hidden node problem. Another section of the graph that requires attention is where the sensor spacing is under 1.2 m. For high AUV speed (0.80 m/s and above), the latency decreases, reaching latency 0 s, which is the same section of the graph as the slightly increased throughput section of Figure 6.24. However, for the side where the AUV speed is low (under 0.80 m/s), the latency is the highest in the graph, reaching 0.13 m/s.

6.3.5 | Power usage

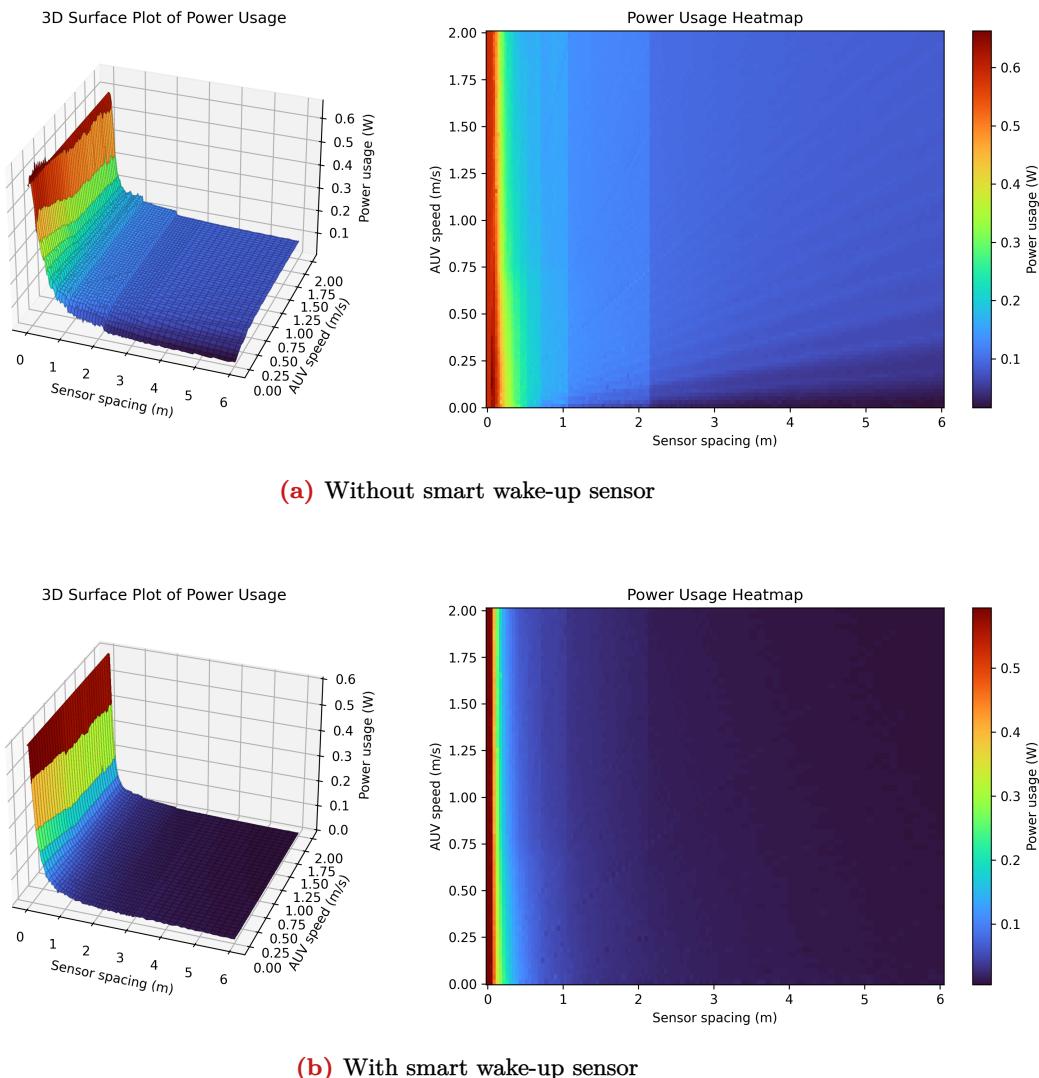


Figure 6.26: Mean power usage STA to AP vs. sensor spacing and AUV speed

The power usage graph without the wake-up sensor reaches maximum mean values of 0.7 W for small sensor spacing of 0 m to 0.2 m and low AUV speed of 0 to 0.40 m/s, but remains high (around 0.6 W) for speeds up to 2 m/s. The power usage significantly reduces for sensor spacing 0.2 to 1 m, with a second drop at 2.1 m. The power usage reaches its minimum for maximum sensor spacing (6 m) and minimum speed (around 0 m/s). This behavior is expected because a higher distance between the sensors means less congestion, thus less number of retransmits and ultimately lower power usage. Moreover, for lower

AUV speed, less numbers of reassociations are needed, which makes the system use less power.

For Figure 6.26b, the additional sleep time decreases the power usage now reaching maximum of 0.6 W again at sensor spacing 0 to 0.2 m and followed by a sudden drop almost to 0 W at around 2.2 m sensor spacing, reaching the minimum power usage at maximum distance between the stations. An important change compared to Figure 6.26a is that AUV speed has much less of an effect on the power usage, both at high and low values. Finally, both graphs follow the patterns identified in Section 6.3.1, highlighting that the hidden node problem affects the power usage as well, which is expected.

6.3.6 | Energy efficiency

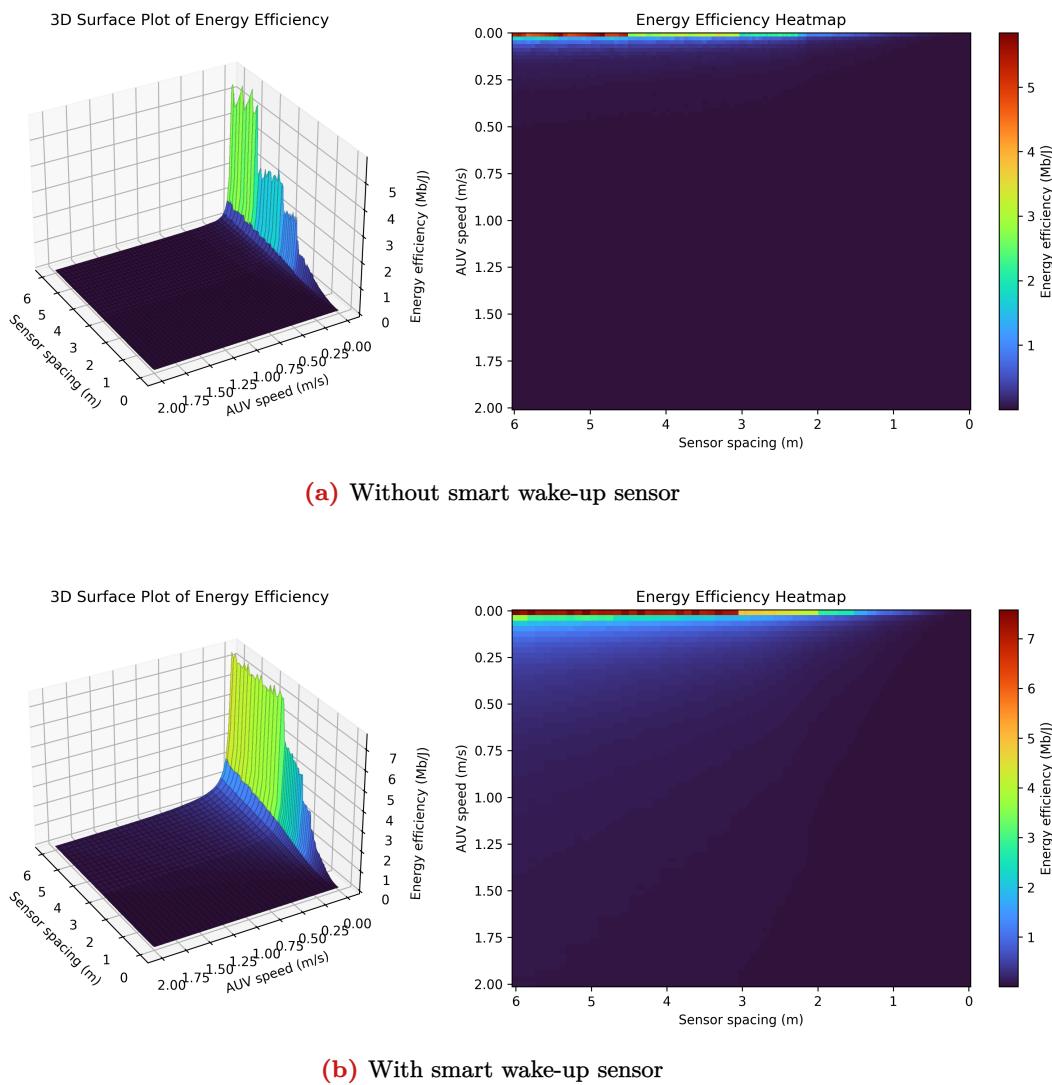


Figure 6.27: Mean energy efficiency STA to AP vs. sensor spacing and AUV speed

Following the trends investigated in the previous subsection in Figure 6.27a the system is most efficient (6 Mb/J) for maximum sensor spacing and minimum AUV speed then it decreases step wise (following the vertical pattern in Section 6.3.1) with the reduction of the distance between sensor and drops significantly for AUV speed of 0.1 m/s or more.

Then, for the graph with the wake-up sensor, we can again see an improvement not only in the value reaching 8 Mb/J but also this efficiency is achieved in more configurations, from sensor spacing 3.2 m to 6 m instead of 4.4 m to 6 m as it is without the sensor. However, for AUV speed greater than 0.25 m/s, the energy efficiency reduces to almost 0. The energy efficiency still follows the hidden node problem patterns, showing the phenomenon's significance on the metric.

6.3.7 | Conclusions

Overall, the graphs show that the best results are for higher sensor spacing and lower AUV speed, with almost all throughput, power usage, and energy efficiency metrics showing significant improvement. Only the latency does not seem to be significantly affected by both metrics, but is still lower at low sensor spacing and higher AUV speed. Also, noting that the wake-up sensor has a significant positive effect on both reducing power usage and improving energy efficiency. Finally, the recurring patterns associated with the hidden node problem provide valuable context for interpreting these results and underline its impact on communication performance.

7 | Conclusions and future work

Wrapping up, this study demonstrated that IEEE 802.11ah (Wi-Fi HaLow) can support high-bandwidth short-range underwater communications with adequate throughput, latency, power consumption, and energy efficiency across a variety of deployment scenarios. Furthermore, our graphs can serve as a reference for future practical work when evaluating the feasibility of using HaLow for a new physical deployment in novel underwater network equipment. Our peer-to-peer simulations revealed that 1 MHz channels allow for longer link ranges than their 2 MHz counterparts, while sacrificing throughput. Across all salinities, latency and power consumption closely tracked link quality, spiking at the operational boundary of the protocol due to the retransmissions before communications failed entirely. In the sensors-to-AP (buoy) topology, we observed the interference patterns that can arise with crowded networks and the effects that they can have on the performance metrics. TCP uncovered surprising effects with larger node counts, a behavior verified by contrasting with UDP results. Lastly, the AUV scenario proves that data muling is one of the key promises of HaLow, providing increased deployment flexibility without sacrificing performance, unlike the sensors-to-AP topology. These findings underscore the importance of medium access configuration and transport-layer dynamics in multi-node underwater networks.

Our work also highlights key implementation and modeling constraints. Improving on the limitations identified in our current study, one avenue for future work is integrating 802.11ah support into more recent ns-3 releases, possibly merging with the main simulator code. The hybrid we employed provided a necessary foundation, but using a newer version would not only streamline compatibility with upgraded PHY modules and improved error models, but also facilitate the use of the expanded TCP variants or antenna improvements as previously mentioned in Section 4.2. Such an effort would likely merit its own dedicated publication, given the involvement with adapting the code and the potential impact on all ns-3 users.

The HaLow implementation we based our study on has limited support for all 802.11ah channel widths and data rates, limiting our testing to the 1 MHz and 2 MHz wide bandwidths. Therefore, we recommend further investigations into the performance of HaLow underwater for channel widths ranging from 4 to 16 MHz.

On the measurement side, an Open Systems Interconnection (OSI) Level 5 probe for FlowMonitor would help capture TCP retransmissions explicitly, revisiting our latency heatmaps to distinguish between true link-edge failures and TCP-induced RTT artifacts. This should clarify the unexpected "latency drop" phenomenon observed at the 10-20 stations mark.

In the peer-to-peer scenario, it needs to be investigated why for salinity of 35 ppt the signal range is not higher than in [12] even though the signal frequency is at least two times smaller. It remains to be determined how the different antenna settings used in [12] affected the results. Furthermore, anomalies in the latency graphs need to be investigated as well.

The Sensors-to-AP scenario needs to be investigated with correct configuration of TIM groups for scenarios with more than 64 stations.

The discrepancy between the total reported power usage and the sum of individual power usages across states should be investigated further. Moreover, not all patterns observed in the per-state power usage graphs are fully understood (particularly, in the SLEEP, RX and CCA_BUSY states), which requires further analysis.

Finally, a completely new scenario and network topology can be investigated, involving multiple "submersed buoys" acting as intermediate nodes. This would allow for a deeper placement of static nodes, enhancing environmental flexibility. This can be supplemented with further water-air or multipath investigations, as "EM waves can cross water-to-air or water-to-earth boundaries easily following the path of least resistance" as reported in [6].

A | Propagation model formulas

The relative permittivity is given by [20]:

$$\epsilon_r = \epsilon_\infty + \frac{\epsilon_s - \epsilon_\infty}{1 + j \frac{f}{f_{ref}}} - \frac{j\sigma}{2\pi f \epsilon_0} \quad (\text{A.1})$$

where ϵ_0 and ϵ_∞ are the static dielectric constants at low and high frequency, respectively. ϵ_∞ is independent of salinity, so $\epsilon_\infty = \epsilon_{water,\infty} = 4.9$. f is the frequency of the signal and f_{ref} is the relaxation frequency $f_{ref} = \frac{1}{2\pi\tau_w}$, where τ_w is the relaxation time, which given the temperature of the environment T ($^{\circ}\text{C}$) and its salinity S (ppt) can be calculated by [20]:

$$\tau_w(T, S) = \tau_w(T, 0) \cdot b(T, S) \quad (\text{A.2})$$

where

$$\tau_w(T, 0) = \frac{1}{2\pi} (1.11 \times 10^{-10} - 3.824 \times 10^{-12}T + 6.938 \times 10^{-14}T^2 - 5.096 \times 10^{-16}T^3) \quad (\text{A.3})$$

$$b(T, S) = 1 + 2.282 \times 10^{-2}T \cdot S - 7.638 \times 10^{-4}S - 7.76 \times 10^{-6}S^2 + 1.105 \times 10^{-8}S^3 \quad (\text{A.4})$$

The static dielectric constant at low frequency (ϵ_s) given the temperature T and salinity S can be calculated by [20]:

$$\epsilon_s(T, S) = \epsilon_s(T, 0) \cdot a(T, S) \quad (\text{A.5})$$

where

$$\begin{aligned} \epsilon_s(T, 0) = & 87.134 - 1.949 \times 10^{-1}T - 1.276 \\ & \times 10^{-2}T^2 + 2.491 \times 10^{-4}T^3 \end{aligned} \quad (\text{A.6})$$

$$a(T, S) = 1 + 1.613 \times 10^{-5}T \cdot S - 3.656 \times 10^{-3}S + 3.21 \times 10^{-5}S^2 - 4232 \times 10^{-7}S^3 \quad (\text{A.7})$$

Given temperature T and salinity S , we calculated the conductivity (σ) of the environment by [47]:

$$\sigma(T, S) = \sigma(25, S) e^{-\phi} \quad (\text{A.8})$$

with

$$\begin{aligned} \phi = & \Delta [0.02033 + 1.266 \times 10^{-4} \Delta + 2.464 \times 10^{-4} \Delta^2 \\ & - S (1.849 \times 10^{-5} - 2.551 \times 10^{-7} \Delta + 2.551 \times 10^{-8} \Delta^2)] \end{aligned} \quad (\text{A.9})$$

$$\Delta = 25 - T, \quad (\text{A.10})$$

$$\sigma(25, S) = S (0.182521 - 1.46192 \times 10^{-3} S + 2.09324 \times 10^{-5} S^2 - 1.28205 \times 10^{-7} S^3) \quad (\text{A.11})$$

With the use of the above mentioned formulas the model is dependent only on temperature and salinity of the environment.

B | RAW configuration results

Table B.1 contains the results from Section 5.1.2, sorted by highest throughput.

Table B.1: RAW configuration results

RAW config (groups, slots)	Throughput (Mbps)	Latency (ns)	Energy usage (J, 20 min)	Energy efficiency (Mb/J)
1 groups, 1 slots	0.0071	44396631	745	6291
1 groups, 67 slots	0.0059	54669155	624	6957
100 groups, 1 slots	0.0064	51583257	736	6137
1 groups, 70 slots	0.0056	63956427	590	6816
1 groups, 4 slots	0.0055	147529574	627	6735
1 groups, 73 slots	0.0049	75010489	588	6585
5 groups, 1 slots	0.0056	104843442	718	5849
1 groups, 76 slots	0.005	106554414	593	6382
1 groups, 79 slots	0.0046	167288813	598	6002
3 groups, 1 slots	0.0051	417950283	698	5567
1 groups, 7 slots	0.0049	716806218	680	6258
5 groups, 4 slots	0.0046	263803340	717	5504
1 groups, 85 slots	0.0041	266782763	611	5672
2 groups, 1 slots	0.006	1341181430	768	5598
1 groups, 82 slots	0.004	351532147	622	5583
5 groups, 7 slots	0.0041	392561175	723	5336
2 groups, 7 slots	0.0046	1010795577	753	5947
1 groups, 88 slots	0.0036	357308716	613	5550
5 groups, 13 slots	0.0039	528700786	718	5357
5 groups, 10 slots	0.0039	476311322	718	5244
2 groups, 4 slots	0.0049	1659588286	754	5804
2 groups, 37 slots	0.004	1187645937	711	5681
2 groups, 10 slots	0.0045	1511145706	749	5675
5 groups, 16 slots	0.0035	750971822	717	5155
2 groups, 13 slots	0.0042	1362522631	745	5613
2 groups, 40 slots	0.004	1270374397	716	5485
1 groups, 16 slots	0.0038	1337669679	680	5763
2 groups, 16 slots	0.0039	1390536683	734	5636
1 groups, 13 slots	0.0038	1305825948	686	5548
2 groups, 31 slots	0.0041	1533377788	721	5623
5 groups, 19 slots	0.0034	831202261	727	4944
2 groups, 22 slots	0.0042	1577072416	721	5519
2 groups, 34 slots	0.004	1476218126	726	5582
2 groups, 43 slots	0.0038	1363896760	714	5435
1 groups, 10 slots	0.0037	1308801600	654	5396
2 groups, 46 slots	0.0039	1611666086	714	5404
2 groups, 28 slots	0.0041	1850302132	735	5460
2 groups, 19 slots	0.004	1855751019	739	5415
1 groups, 91 slots	0.0031	1028938033	635	4775

Continued on the next page

Table B.1 – *continued from previous page*

RAW config (groups, slots)	Throughput (Mbps)	Latency (ns)	Energy usage (J, 20 min)	Energy efficiency (Mb/J)
2 groups, 25 slots	0.0037	1677609276	734	5348
1 groups, 19 slots	0.0032	1649521622	680	5017
1 groups, 22 slots	0.0032	1733768896	666	5097
2 groups, 49 slots	0.0035	1890115446	731	4630
1 groups, 94 slots	0.0026	1473633449	642	4350
1 groups, 97 slots	0.0025	1803147352	648	4042
1 groups, 25 slots	0.0027	2158477871	675	4393
3 groups, 4 slots	0.003	2609991569	719	4333
1 groups, 34 slots	0.0027	2462890904	676	4239
1 groups, 52 slots	0.0026	2548752734	666	4425
3 groups, 10 slots	0.0045	3316262548	733	3945
1 groups, 28 slots	0.0025	2418580547	670	4127
1 groups, 31 slots	0.0025	2585691894	679	4035
1 groups, 37 slots	0.0024	2576114538	677	3995
1 groups, 40 slots	0.0025	2719280630	685	4083
1 groups, 49 slots	0.0025	2785945765	668	4143
3 groups, 13 slots	0.0027	3077495818	718	4129
3 groups, 7 slots	0.0028	3126779436	732	4050
1 groups, 55 slots	0.0025	3075297506	679	4127
1 groups, 100 slots	0.0021	2600531514	651	3599
3 groups, 22 slots	0.0026	3235779188	719	3975
3 groups, 28 slots	0.0026	3309807404	724	3985
1 groups, 46 slots	0.0024	3144724177	678	3798
3 groups, 19 slots	0.0025	3489515786	746	3820
1 groups, 43 slots	0.0021	3220451530	684	3495
3 groups, 31 slots	0.0025	3536727582	724	3765
3 groups, 25 slots	0.0024	3542170948	726	3703
3 groups, 16 slots	0.0024	3703402378	746	3814
3 groups, 34 slots	0.0025	3684784912	740	3732
1 groups, 58 slots	0.0022	3538722922	687	3703
1 groups, 61 slots	0.0019	3765144479	679	3139

C | Individual contributions

C.1 | Deins Kovalcuks

Helped research, summarize and shortlist proposals for simulation scenarios used in this study.

Helped identify a relevant path loss formula.

Helped process logs by generating visualizations.

Verified that the configured MCS (Modulation and Coding Scheme) was used correctly in the simulation.

Analyzed and verified data in the Sensor-to-AP scenario, noticing issues with TIM configuration, and observing bugs in the source code that failed to catch the misconfiguration at runtime.

C.2 | Zofia Rynkiewicz

In the beginning, I helped finalize the scenarios for the implementation of the simulations and also looked for potential use cases for which our work could be useful. I have researched potential performance metrics that we could investigate. I have worked on the final formulas for the path loss that we could use for the model for underwater communication, as Aqua-Sim turned out not to be the most feasible solution. I also conducted extensive research on the properties of water to understand how it affects the path loss and what assumptions we should use in each scenario. Lastly, I have contributed a lot to the report, set it to provide the first draft for most of the sections, and changed them accordingly to the feedback of my colleagues.

C.3 | Theodore Visoiu

I helped configure the ns-3 simulations and part of the server infrastructure they ran on, and I developed most of the initial visualization tools and support scripts for large-scale simulation management. I also leveraged my ns-3 expertise to add re-association support at the MAC layer, and I implemented the majority of our logging and telemetry systems for the resulting graphs. Additionally, I guided my colleagues through ns-3's architecture and limitations, enabling them to help me identify relevant literature on protocol quirks, such as the inherent unfairness in 802.11ah [31], which contributed to our TCP vs UDP investigations in Section 6.2. I also helped out by reviewing this report and fixing L^AT_EX issues.

C.4 | Maxim Yordanov

At the start of the project, I took part in finalizing the scenarios we wanted to test and helped identify possible use cases for the WiFi HaLow protocol in underwater communication. I then researched the underwater propagation model and helped finalize the formulas for path loss and propagation delay. After that, I worked on implementing the formulas into the ns-3 simulator and contributed to setting up and running the simulation scenarios. I was involved in generating some of the visualizations and took part in analyzing and interpreting the simulation results. Lastly, I mainly worked on the Data Mulling results and was responsible for writing the related reasoning and conclusions in the final report.

References

- [1] G. A. López-Ramírez and A. Aragón-Zavala, "Wireless sensor networks for water quality monitoring: A comprehensive review," *IEEE Access*, vol. 11, pp. 95 120–95 142, 2023, ISSN: 2169-3536. DOI: [10.1109/ACCESS.2023.3308905](https://doi.org/10.1109/ACCESS.2023.3308905). [Online]. Available: <https://ieeexplore.ieee.org/document/10231325/> (visited on 04/20/2025).
- [2] *Ns-3 network simulator*, version 3.23, 3.25. [Online]. Available: nsnam.org (visited on 04/16/2025).
- [3] J. Heidemann, M. Stojanovic, and M. Zorzi, "Underwater sensor networks: Applications, advances and challenges," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 370, no. 1958, pp. 158–175, Jan. 13, 2012, ISSN: 1364-503X, 1471-2962. DOI: [10.1098/rsta.2011.0214](https://doi.org/10.1098/rsta.2011.0214). [Online]. Available: <https://royalsocietypublishing.org/doi/10.1098/rsta.2011.0214> (visited on 04/20/2025).
- [4] B. Pierre-Jean, F. Philippe, T. Beatrice, *et al.*, "Contactless data transfer for autonomous underwater vehicle docking station," in *OCEANS 2021: San Diego – Porto*, San Diego, CA, USA: IEEE, Sep. 20, 2021, pp. 1–5, ISBN: 978-0-692-93559-0. DOI: [10.23919/OCEANS44145.2021.9705698](https://doi.org/10.23919/OCEANS44145.2021.9705698). [Online]. Available: <https://ieeexplore.ieee.org/document/9705698/> (visited on 04/20/2025).
- [5] R. Alahmad, H. Alraie, R. Hasaba, *et al.*, "Performance analysis of underwater radiofrequency communication in seawater: An experimental study," *Journal of Marine Science and Engineering*, vol. 12, no. 11, p. 2104, Nov. 20, 2024, ISSN: 2077-1312. DOI: [10.3390/jmse12112104](https://doi.org/10.3390/jmse12112104). [Online]. Available: <https://www.mdpi.com/2077-1312/12/11/2104> (visited on 04/20/2025).
- [6] X. Che, I. Wells, G. Dickers, P. Kear, and X. Gong, "Re-evaluation of RF electromagnetic communication in underwater sensor networks," *IEEE Communications Magazine*, vol. 48, no. 12, pp. 143–151, Dec. 2010, ISSN: 0163-6804. DOI: [10.1109/MCOM.2010.5673085](https://doi.org/10.1109/MCOM.2010.5673085). [Online]. Available: [http://ieeexplore.ieee.org/document/5673085/](https://ieeexplore.ieee.org/document/5673085/) (visited on 04/27/2025).
- [7] A. Dala and T. Arslan, "Design, implementation, and measurement procedure of underwater and water surface antenna for LoRa communication," *Sensors*, vol. 21, no. 4, p. 1337, Feb. 13, 2021, ISSN: 1424-8220. DOI: [10.3390/s21041337](https://doi.org/10.3390/s21041337). [Online]. Available: <https://www.mdpi.com/1424-8220/21/4/1337> (visited on 04/20/2025).
- [8] F. Teixeira, P. Freitas, L. Pessoa, R. Campos, and M. Ricardo, "Evaluation of IEEE 802.11 underwater networks operating at 700 MHz, 2.4 GHz and 5 GHz," in *Proceedings of the International Conference on Underwater Networks %Systems - WUWNET '14*, Rome, Italy: ACM Press, 2014, pp. 1–5, ISBN: 978-1-4503-3277-4. DOI: [10.1145/2671490.2674571](https://doi.org/10.1145/2671490.2674571). [Online]. Available: <http://dl.acm.org/citation.cfm?doid=2671490.2674571> (visited on 04/20/2025).
- [9] M. Alam, N. Ahmed, R. Matam, and F. A. Barbhuiya, "Analyzing the suitability of IEEE 802.11ah for next generation internet of things: A comparative study," *Ad Hoc Networks*, vol. 156, p. 103 437, Apr. 2024, ISSN: 15708705. DOI: [10.1016/j.adhoc.2024.103437](https://doi.org/10.1016/j.adhoc.2024.103437). [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1570870524000489> (visited on 04/20/2025).
- [10] S. Aust, R. V. Prasad, and I. G. M. M. Niemegeers, "IEEE 802.11ah: Advantages in standards and further challenges for sub 1 GHz wi-fi," in *2012 IEEE International Conference on Communications (ICC)*, Ottawa, ON, Canada: IEEE, Jun. 2012, pp. 6885–6889, ISBN: 978-1-4577-2053-6. DOI: [10.1109/ICC.2012.6364903](https://doi.org/10.1109/ICC.2012.6364903). [Online]. Available: [http://ieeexplore.ieee.org/document/6364903/](https://ieeexplore.ieee.org/document/6364903/) (visited on 04/20/2025).

- [11] S. Maudet, G. Andrieux, R. Chevillon, and J.-F. Diouris, “Practical evaluation of wi-fi HaLow performance,” *Internet of Things*, vol. 24, p. 100957, Dec. 2023, ISSN: 25426605. DOI: [10.1016/j.iot.2023.100957](https://doi.org/10.1016/j.iot.2023.100957). [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S2542660523002809> (visited on 04/20/2025).
- [12] J. C. R. Guerrero, I. B. Mabrouk, M. Alhassan, M. Nedil, and T. Ciamulski, “ON THE PATH LOSS MODEL FOR 5-GHZ MICROWAVE-BASED PINLESS SUBSEA CONNECTORS,” *Progress In Electromagnetics Research Letters*, vol. 82, pp. 147–153, 2019, ISSN: 1937-6480. DOI: [10.2528/PIERL18102705](https://doi.org/10.2528/PIERL18102705). [Online]. Available: <http://www.jpier.org/PIERL/pier.php?paper=18102705> (visited on 04/20/2025).
- [13] P. S. S. P. Ganesh and H. Venkataraman, “RF-based wireless communication for shallow water networks: Survey and analysis,” *Wireless Personal Communications*, vol. 120, no. 4, pp. 3415–3441, Oct. 2021, ISSN: 0929-6212, 1572-834X. DOI: [10.1007/s11277-021-09068-w](https://doi.org/10.1007/s11277-021-09068-w). [Online]. Available: <https://link.springer.com/10.1007/s11277-021-09068-w> (visited on 04/20/2025).
- [14] K. Li, C.-C. Shen, and G. Chen, “Energy-constrained bi-objective data muling in underwater wireless sensor networks,” in *The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS 2010)*, San Francisco, CA, USA: IEEE, Nov. 2010, pp. 332–341, ISBN: 978-1-4244-7488-2. DOI: [10.1109/MASS.2010.5664026](https://doi.org/10.1109/MASS.2010.5664026). [Online]. Available: <http://ieeexplore.ieee.org/document/5664026/> (visited on 04/20/2025).
- [15] Y. Guo, F. Kong, D. Zhu, A. S. Tosun, and Q. Deng, “Sensor placement for lifetime maximization in monitoring oil pipelines,” in *Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems*, Stockholm Sweden: ACM, Apr. 13, 2010, pp. 61–68, ISBN: 978-1-4503-0066-7. DOI: [10.1145/1795194.1795204](https://doi.org/10.1145/1795194.1795204). [Online]. Available: <https://dl.acm.org/doi/10.1145/1795194.1795204> (visited on 04/20/2025).
- [16] M. A. Akkaş and R. Sokullu, “Channel modeling and analysis for wireless underground sensor networks in water medium using electromagnetic waves in the 300–700 MHz range,” *Wireless Personal Communications*, vol. 84, no. 2, pp. 1449–1468, Sep. 2015, ISSN: 0929-6212, 1572-834X. DOI: [10.1007/s11277-015-2697-3](https://doi.org/10.1007/s11277-015-2697-3). [Online]. Available: <http://link.springer.com/10.1007/s11277-015-2697-3> (visited on 06/30/2025).
- [17] S. Sendra, J. Lloret, J. J. P. C. Rodrigues, and J. M. Aguiar, “Underwater wireless communications in freshwater at 2.4 GHz,” *IEEE Communications Letters*, vol. 17, no. 9, pp. 1794–1797, Sep. 2013, ISSN: 1089-7798. DOI: [10.1109/LCOMM.2013.072313.131214](https://doi.org/10.1109/LCOMM.2013.072313.131214). [Online]. Available: <http://ieeexplore.ieee.org/document/6574941/> (visited on 06/30/2025).
- [18] S. S. Godkhindi, P. Reddy, and T. V. Prabhakar, “Link quality analysis of LoRa for internet of underwater things,” in *2023 15th International Conference on COMmunication Systems & NETworkS (COMSNETS)*, Bangalore, India: IEEE, Jan. 3, 2023, pp. 222–224, ISBN: 978-1-6654-7706-2. DOI: [10.1109/COMSNETS56262.2023.10041316](https://doi.org/10.1109/COMSNETS56262.2023.10041316). [Online]. Available: <https://ieeexplore.ieee.org/document/10041316/> (visited on 06/30/2025).
- [19] K. Lin, T. Hao, W. Zheng, and W. He, “Analysis of LoRa link quality for underwater wireless sensor networks: A semi-empirical study,” in *2019 IEEE Asia-Pacific Microwave Conference (APMC)*, Singapore, Singapore: IEEE, Dec. 2019, pp. 120–122, ISBN: 978-1-7281-3517-5. DOI: [10.1109/APMC46564.2019.9038666](https://doi.org/10.1109/APMC46564.2019.9038666). [Online]. Available: <https://ieeexplore.ieee.org/document/9038666/> (visited on 04/20/2025).

- [20] G. Hattab, M. El-Tarhuni, M. Al-Ali, T. Joudeh, and N. Qaddoumi, “An underwater wireless sensor network with realistic radio frequency path loss model,” *International Journal of Distributed Sensor Networks*, vol. 9, no. 3, p. 508 708, Mar. 1, 2013, ISSN: 1550-1477, 1550-1477. DOI: [10.1155/2013/508708](https://doi.org/10.1155/2013/508708). [Online]. Available: <http://journals.sagepub.com/doi/10.1155/2013/508708> (visited on 04/20/2025).
- [21] P. G. Pss and N. Lavanya, “Investigation of RF-based networking for underwater wireless sensor networks using dynamic cluster head selection strategy,” in *2023 11th International Symposium on Electronic Systems Devices and Computing (ESDC)*, Sri City, India: IEEE, May 4, 2023, pp. 1–5, ISBN: 978-1-6654-5572-5. DOI: [10.1109/ESDC56251.2023.10149879](https://doi.org/10.1109/ESDC56251.2023.10149879). [Online]. Available: <https://ieeexplore.ieee.org/document/10149879/> (visited on 06/30/2025).
- [22] M. Mostafa, H. Esmaiel, and E. M. Mohamed, “A comparative study on underwater communications for enabling c/u plane splitting based hybrid UWSNs,” in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, Barcelona: IEEE, Apr. 2018, pp. 1–6, ISBN: 978-1-5386-1734-2. DOI: [10.1109/WCNC.2018.8377209](https://doi.org/10.1109/WCNC.2018.8377209). [Online]. Available: <https://ieeexplore.ieee.org/document/8377209/> (visited on 06/30/2025).
- [23] F. B. Teixeira, N. Moreira, R. Campos, and M. Ricardo, “Data muling approach for long-range broadband underwater communications,” in *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Barcelona, Spain: IEEE, Oct. 2019, pp. 1–4, ISBN: 978-1-7281-3316-4. DOI: [10.1109/WiMOB.2019.8923472](https://doi.org/10.1109/WiMOB.2019.8923472). [Online]. Available: <https://ieeexplore.ieee.org/document/8923472/> (visited on 06/30/2025).
- [24] H. Zheng, N. Wang, and J. Wu, “Minimizing deep sea data collection delay with autonomous underwater vehicles,” *Journal of Parallel and Distributed Computing*, vol. 104, pp. 99–113, Jun. 2017, ISSN: 07437315. DOI: [10.1016/j.jpdc.2017.01.006](https://doi.org/10.1016/j.jpdc.2017.01.006). [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0743731517300126> (visited on 07/05/2025).
- [25] P. K. Donta, B. S. P. Rao, T. Amgoth, C. S. R. Annavarapu, and S. Swain, “Data collection and path determination strategies for mobile sink in 3d WSNs,” *IEEE Sensors Journal*, vol. 20, no. 4, pp. 2224–2233, Feb. 15, 2020, ISSN: 1530-437X, 1558-1748, 2379-9153. DOI: [10.1109/JSEN.2019.2949146](https://doi.org/10.1109/JSEN.2019.2949146). [Online]. Available: <https://ieeexplore.ieee.org/document/8882255/> (visited on 07/05/2025).
- [26] P. K. D., T. Amgoth, and C. S. R. Annavarapu, “ACO-based mobile sink path determination for wireless sensor networks under non-uniform data constraints,” *Applied Soft Computing*, vol. 69, pp. 528–540, Aug. 2018, ISSN: 15684946. DOI: [10.1016/j.asoc.2018.05.008](https://doi.org/10.1016/j.asoc.2018.05.008). [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1568494618302692> (visited on 07/05/2025).
- [27] L. Tao, X. M. Zhang, and W. Liang, “Efficient algorithms for mobile sink aided data collection from dedicated and virtual aggregation nodes in energy harvesting wireless sensor networks,” *IEEE Transactions on Green Communications and Networking*, vol. 3, no. 4, pp. 1058–1071, Dec. 2019, ISSN: 2473-2400. DOI: [10.1109/TGCN.2019.2927619](https://doi.org/10.1109/TGCN.2019.2927619). [Online]. Available: <https://ieeexplore.ieee.org/document/8758390/> (visited on 07/05/2025).
- [28] Q. Fu, A. Song, F. Zhang, and M. Pan, “Reinforcement learning-based trajectory optimization for data muling with underwater mobile nodes,” *IEEE Access*, vol. 10, pp. 38 774–38 784, 2022, ISSN: 2169-3536. DOI: [10.1109/ACCESS.2022.3165046](https://doi.org/10.1109/ACCESS.2022.3165046). [Online]. Available: <https://ieeexplore.ieee.org/document/9749248/> (visited on 07/05/2025).
- [29] *ChatGPT*, version 4o. [Online]. Available: chat.openai.com (visited on 04/16/2025).

- [30] L. Tian, A. Šljivo, S. Santi, E. De Poorter, J. Hoobeke, and J. Famaey, “Extension of the IEEE 802.11ah ns-3 simulation module,” in *Proceedings of the 10th Workshop on ns-3 - WNS3 '18*, Surathkal, India: ACM Press, 2018, pp. 53–60, ISBN: 978-1-4503-6413-3. DOI: [10.1145/3199902.3199906](https://doi.org/10.1145/3199902.3199906). [Online]. Available: <http://dl.acm.org/citation.cfm?doid=3199902.3199906> (visited on 04/20/2025).
- [31] R. Verhoeven, S. Kempinski, and N. Meratnia, “Performance evaluation of wi-fi HaLow, NB-IoT and LoRa for smart city applications,” in *Proceedings of the 19th ACM International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks on 19th ACM International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks*, Montreal Quebec Canada: ACM, Oct. 24, 2022, pp. 17–24, ISBN: 978-1-4503-9483-3. DOI: [10.1145/3551663.3558596](https://doi.org/10.1145/3551663.3558596). [Online]. Available: <https://dl.acm.org/doi/10.1145/3551663.3558596> (visited on 04/20/2025).
- [32] H. Wu, S. Nabar, and R. Poovendran, “An energy framework for the network simulator 3 (NS-3),” in *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*, ser. SIMUTools '11, event-place: Barcelona, Spain, Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2011, pp. 222–230, ISBN: 978-1-936968-00-8. DOI: [10.5555/2151054.2151096](https://doi.org/10.5555/2151054.2151096).
- [33] D. Halperin, B. Greenstein, A. Sheth, and D. Wetherall, “Demystifying 802.11n power consumption,” in *Proceedings of the 2010 International Conference on Power Aware Computing and Systems*, ser. HotPower'10, event-place: Vancouver, BC, Canada, USA: USENIX Association, 2010, p. 1. DOI: [10.5555/1924920.1924928](https://doi.org/10.5555/1924920.1924928).
- [34] H. Friis, “A note on a simple transmission formula,” *Proceedings of the IRE*, vol. 34, no. 5, pp. 254–256, May 1946, ISSN: 0096-8390. DOI: [10.1109/JRPROC.1946.234568](https://doi.org/10.1109/JRPROC.1946.234568). [Online]. Available: <http://ieeexplore.ieee.org/document/1697062/> (visited on 04/20/2025).
- [35] R. Martin, S. Rajasekaran, and Z. Peng, “Aqua-sim next generation: An NS-3 based underwater sensor network simulator,” in *Proceedings of the International Conference on Underwater Networks & Systems*, Halifax NS Canada: ACM, Nov. 6, 2017, pp. 1–8, ISBN: 978-1-4503-5561-2. DOI: [10.1145/3148675.3148679](https://doi.org/10.1145/3148675.3148679). [Online]. Available: <https://dl.acm.org/doi/10.1145/3148675.3148679> (visited on 04/20/2025).
- [36] Linux Wireless Documentation Project, *Minstrel rate control algorithm*, version mac80211, Type: Documentation, 2025. [Online]. Available: <https://wireless.docs.kernel.org/en/latest/en/developers/documentation/mac80211/ratecontrol/minstrel.html>.
- [37] National Oceanic and Atmospheric Administration. “Sea water,” Sea Water — National Oceanic and Atmospheric Administration. (Mar. 28, 2023), [Online]. Available: <https://www.noaa.gov/jetstream/ocean/sea-water> (visited on 04/20/2025).
- [38] A. V. Mishonov, “World ocean database 2023,” 2024, Publisher: NOAA National Centers for Environmental Information. DOI: [10.25923/Z885-H264](https://doi.org/10.25923/Z885-H264). [Online]. Available: <https://repository.library.noaa.gov/view/noaa/66204> (visited on 04/20/2025).
- [39] W. Musie and G. Gonfa, “Fresh water resource, scarcity, water salinity challenges and possible remedies: A review,” *Heliyon*, vol. 9, no. 8, e18685, Aug. 2023, ISSN: 24058440. DOI: [10.1016/j.heliyon.2023.e18685](https://doi.org/10.1016/j.heliyon.2023.e18685). [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S2405844023058930> (visited on 04/20/2025).

- [40] I. Cappelli, A. Fort, M. Mugnaini, S. Parrino, and A. Pozzebon, “Underwater to above water LoRaWAN networking: Theoretical analysis and field tests,” *Measurement*, vol. 196, p. 111 140, Jun. 2022, ISSN: 02632241. DOI: [10.1016/j.measurement.2022.111140](https://doi.org/10.1016/j.measurement.2022.111140). [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0263224122004006> (visited on 04/20/2025).
- [41] A. Pozzebon, I. Cappelli, F. Campagnaro, R. Francescon, and M. Zorzi, “LoRaWAN transmissions in salt water for superficial marine sensor networking: Laboratory and field tests,” *Sensors*, vol. 23, no. 10, p. 4726, May 13, 2023, ISSN: 1424-8220. DOI: [10.3390/s23104726](https://doi.org/10.3390/s23104726). [Online]. Available: <https://www.mdpi.com/1424-8220/23/10/4726> (visited on 04/24/2025).
- [42] *SolidWorks*, version 2022 SP02. [Online]. Available: [solidworks.com](https://www.solidworks.com).
- [43] Ø. Hasvold, T. Lian, E. Haakaas, N. Størkersen, O. Perelman, and S. Cordier, “CLIPPER: A long-range, autonomous underwater vehicle using magnesium fuel and oxygen from the sea,” *Journal of Power Sources*, vol. 136, no. 2, pp. 232–239, Oct. 2004, ISSN: 03787753. DOI: [10.1016/j.jpowsour.2004.03.023](https://doi.org/10.1016/j.jpowsour.2004.03.023). [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0378775304002873> (visited on 04/22/2025).
- [44] E. H. Kolås, T. Mo-Bjørkelund, and I. Fer, “Technical note: Turbulence measurements from a light autonomous underwater vehicle,” *Ocean Science*, vol. 18, no. 2, pp. 389–400, Mar. 24, 2022, ISSN: 1812-0792. DOI: [10.5194/os-18-389-2022](https://doi.org/10.5194/os-18-389-2022). [Online]. Available: <https://os.copernicus.org/articles/18/389/2022/> (visited on 04/22/2025).
- [45] J. C. Reyes-Guerrero and T. Ciamulski, “Influence of temperature on signal attenuation at microwaves frequencies underwater,” in *OCEANS 2015 - Genova*, Genova, Italy: IEEE, May 2015, pp. 1–4, ISBN: 978-1-4799-8736-8. DOI: [10.1109/OCEANS-Genova.2015.7271562](https://doi.org/10.1109/OCEANS-Genova.2015.7271562). [Online]. Available: <http://ieeexplore.ieee.org/document/7271562/> (visited on 04/24/2025).
- [46] A. Šljivo, D. Kerkhove, L. Tian, *et al.*, “Performance evaluation of IEEE 802.11ah networks with high-throughput bidirectional traffic,” *Sensors*, vol. 18, no. 2, p. 325, Jan. 23, 2018, ISSN: 1424-8220. DOI: [10.3390/s18020325](https://doi.org/10.3390/s18020325). [Online]. Available: <https://www.mdpi.com/1424-8220/18/2/325> (visited on 04/29/2025).
- [47] A. Stogryn, “Equations for calculating the dielectric constant of saline water (correspondence),” *IEEE Transactions on Microwave Theory and Techniques*, vol. 19, no. 8, pp. 733–736, Aug. 1971, ISSN: 0018-9480. DOI: [10.1109/TMTT.1971.1127617](https://doi.org/10.1109/TMTT.1971.1127617). [Online]. Available: <http://ieeexplore.ieee.org/document/1127617/> (visited on 04/20/2025).