# NetSim®

Accelerate Network R & D

# Underwater Acoustic Networks (UWAN)

A Network Simulation & Emulation Software

By

TETCOS LLP

**Contact us at**
TETCOS LLP
# 214, 39th A Cross, 7th Main, 5th Block Jayanagar,
Bangalore - 560 041, Karnataka, INDIA.
Phone: +91 80 26630624
E-Mail: sales@tetcos.com
Visit: www.tetcos.com

## Table of Contents

# 1 Introduction

Water covers 71% of the Earth. Underwater communication is essential for a wide assortment of applications covering defence, environmental monitoring, commercial exploration, and scientific discovery. Severe attenuation in water limits the range of electromagnetic, optical and magnetic induction-based communications to just a few meters, leaving acoustic communications as the de facto means for wireless data transfer across tens of kilometers.

NetSim's UWAN library enables users to design, simulate and analyze performance of underwater networks that use acoustic communication.

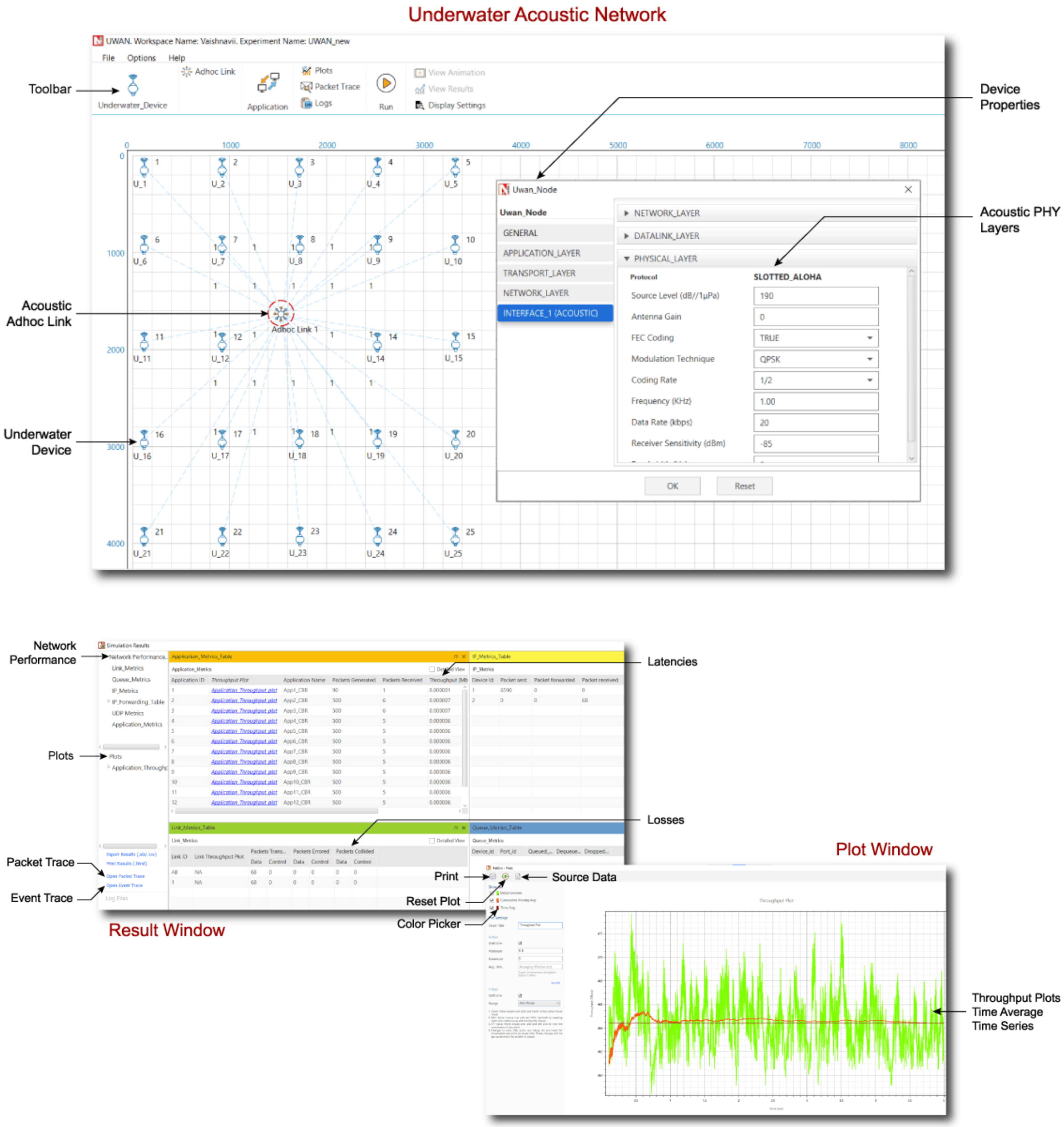



Figure 1-1: NetSim's UWAN design window, the results dashboard, and the plots window

NetSim UWAN simulations are full stack with all 5 layers of the TCP/IP stack being supported as explained below:

- L5, Application: Users can model various kinds of applications as explained in section 3.4. This library supports a new (and default) underwater sensor application - with small packet sizes (order of 10s of Bytes), and large inter-packet arrival times (orders of seconds) - that parallels typical underwater applications.
- L4, Transport: UDP Protocol is supported. TCP is not provided as an option since it is not used in UWAN applications, due to the very low communication bit rates and the high propagation delays.
- L3, Network: Static routing is supported.
- L2, Data Link: Slotted-aloha protocol is supported.
- L1, Physical: Specialized underwater acoustic PHY model and the Thorp propagation model, are supported. Omni directional antennas are assumed.

UWAN is architected to interface with NetSim component 2 (Legacy networks) which provides L2 functionality and component 3 (Advanced switching and routing) which provides the L3 static routing functionality.

The UWAN library is available as Component 12 and is currently available only in NetSim Standard and NetSim Pro versions. Protocol source C code is open to users; it is modular and customizable to help researchers to design and test their own UWAN protocols.

# 2 Simulation GUI

## 2.1 Create Scenario

Open NetSim and click **New Simulation** → **Underwater Acoustic Networks** as shown Figure 2-1.

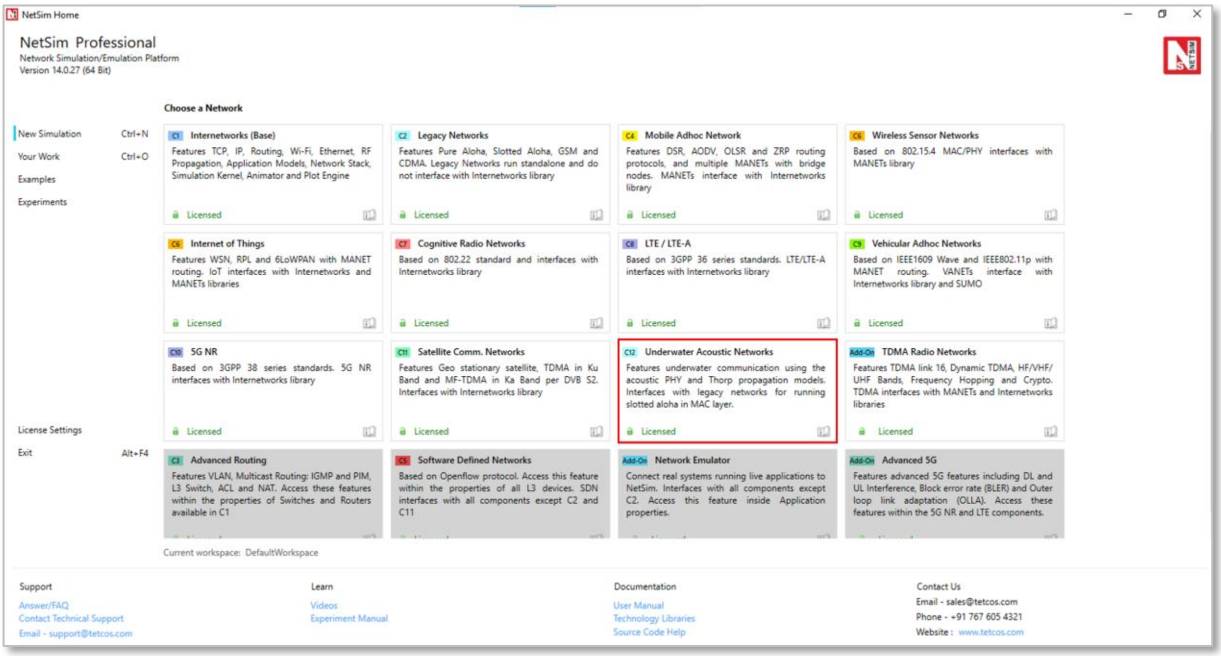



Figure 2-1: NetSim Home Screen

## 2.2 Devices specific to NetSim UWAN Library

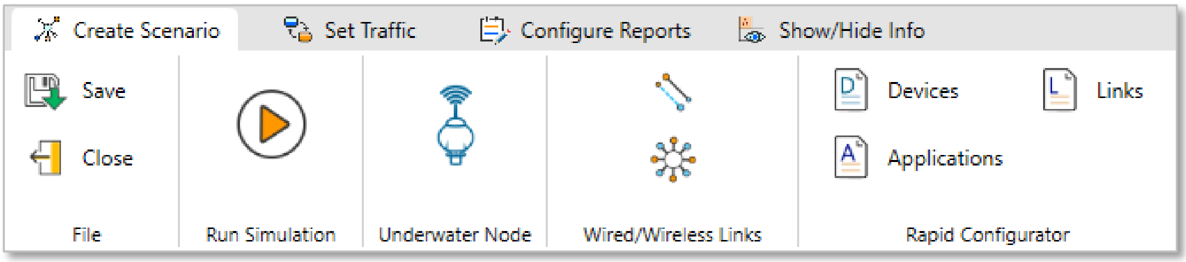



Figure 2-2: The Devices present in the ribbon of NetSim GUI

## 2.3 Placement of devices on the grid environment

Add an Underwater Device - Click on the **Underwater_Device** icon on the toolbar and place the device in the grid.

## 2.4 GUI Configuration Parameters

The UWAN parameters can be accessed by right clicking on an Underwater device and selecting Interface (Acoustic) Properties → Physical Layers.

| UWAN Properties | | | |
|---|---|---|---|
| **Interface (Acoustic) – Physical Layer** | | | |
| **Parameter** | **Type** | **Range** | **Description** |
| Source Level | Local | 170 - 225 $dB//$ $1\ \mu Pa$<br><br>Default: 190 | It is the signal level of the transmitter. For converting from electrical transmit power, $P_{tx}^{el}$ in Watts (W), to SL in dB// 1 µPa, the following equation can be used $SL = 10\log_{10}(P_{tx}^{el}\,[W]) + 170.8$ |
| Antenna Gain (dBi) | Local | -1000 to 1000 dBi | A relative measure of an antenna's ability to direct or concentrate radio frequency energy in a particular direction or pattern. The measurement is typically measured in dBi. |
| Forward error correction coding (FEC) | Fixed | TRUE | FEC is used for controlling Error-Correcting code, in data over unreliable or noisy communication channels. This is always set to true and the SNR BER calculations factor in FEC. |
| Modulation | Local | QPSK<br>BPSK<br>FSK<br>16QAM<br>64QAM<br>256QAM | Modulation is the process of varying one waveform in relation to another waveform. It is used to transfer data over an analog channel. |
| Coding Rate | Local | 1/2, 2/3, 3/4, 5/6 | It states what portion of the total amount of information is useful (non-redundant). This code rate is typically a fractional number. |
| Frequency | Local | 0.01-1000kHz | The centre frequency in the transmission bandwidth |
| Data rate (kbps) | Local | 0-255 kbps | It is the number of kilo-bits that are conveyed or processed per second |
| Receiver Sensitivity (dBm) | Local | -120 – 0dBm<br>Default: -85 dBm | It is the lowest power level at which the receiver can detect the acoustic signal and demodulate data.<br>The interference threshold is equal to the receive sensitivity as explained in 3.1.8 |
| Bandwidth (Hz) | Local | 0-1000 Hz | Bandwidth is the range of frequencies occupied by acoustic signals. |

Table 2-1: UWAN Config Properties

# 3 Model Features

## 3.1 Acoustic PHY

Underwater acoustic channels are generally recognized as one of the most difficult communication media in use today. The worst properties of radio channels - poor link quality, and high latency - are combined in the acoustic channel. Acoustic propagation is best supported at low frequencies and is characterized by two major factors: attenuation that increases with signal frequency, and the low speed of sound (≈1500 $m/s$).

### 3.1.1 Speed of sound

The propagation delay model is complex due to the dependency of the speed of sound on the depth of the water. The speed of sound in water, in meters per second, is given by the formula

$$c_{sound} = 1449.05 + 45.7t - 5.21 \times t^2 + 0.23 \times t^3 + (1.333 - 0.126t + 0.009 \times t^2)(S - 35) + 16.3 \times z + 0.18 \times z^2$$

Where $t$ is one-tenth of the temperature of the water in degrees Celsius, $z$ is the average depth in km of the temperature zone and $S$ is the salinity of the water in parts per thousand.

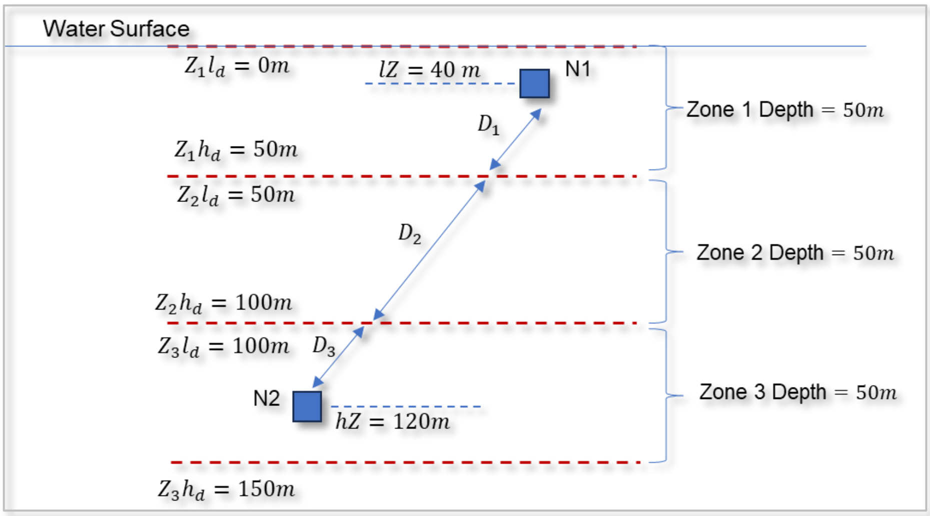



Figure 3-1: An illustration of temperature zones in NetSim where sensor N1 is in zone 1 and sensor N2 is in zone 3.

In the figure, $l_d$ denotes the starting depth and $h_d$ denotes the ending depth of each zone. The zone count (three in this case) and the depth of each zone can be set in NetSim. The average depth in Zone 1 is $\frac{Z_1 l_d + Z_1 h_d}{2} = \frac{0+50}{2} = 25m$, in Zone 2 is $\frac{Z_2 l_d + Z_2 h_d}{2} = \frac{50+100}{2} = 75m$, and in zone 3 is $\frac{Z_3 l_d + Z_3 h_d}{2} = \frac{100+150}{2} = 125m$. The propagation delay is computed separately as

$$\Delta_{tot} = \Delta_1 + \Delta_2 + \Delta_3 = \frac{D_1}{C_{sound}^{zone1}} + \frac{D_2}{C_{sound}^{zone2}} + \frac{D_3}{C_{sound}^{zone3}}$$

### 3.1.2 Transmit power and Source Level

An acoustic signal propagates as a pressure wave, whose power is measured in Pascals (commonly, in dB relative to a micro-Pascal). NetSim GUI takes source level, $SL$, as the input, with units $dB// 1\ \mu Pa$. We may think of $dB// 1\ \mu Pa$ in acoustics as playing the role of $dBm$ in RF. When referring to the Tx/Rx (acoustic) power it is customary to specify the acoustic intensity value in decibel (dB) relative to the intensity due to 1 micro-pascal ($\mu Pa$) root mean square (RMS) pressure. In seawater, 1 W of radiated acoustic power creates a sound field of intensity 170.8 $dB\ //\ 1\ \mu Pa$, 1m away from the source. For converting from electrical transmit power, $P_{tx}^{el}$ in Watts (W), to $SL$ in $dB// 1\ \mu Pa$, the following equation can be used

$$SL = 10 \log_{10}(\xi \times P_{tx}^{el}\ [W]) + 170.8$$

where $\xi$ is transducer efficiency. The 170.8 dB accounts for the conversion between $dB// 1\ \mu Pa$ and $W$. Given below in Table 3-1 are the properties of six commercial and research underwater modems [1]. $SL$ is computed assuming $\xi = 1$.

| Underwater modem | $P_{tx}^{el}\ [W]$ | $SL[dB//1\ \mu Pa]$ | $f\ [kHz]$ | $Data\ Rate\ [kbps]$ |
|---|---|---|---|---|
| EvoLogics S2CR 18/34 WiSE | 35 | 186.2407 | 26 | 13.90 |
| WHOI Micromodem | 48 | 187.6124 | 25 | 5 |
| Teledyne Benthos ATM9XX | 20 | 183.8103 | 24.50 | 15.36 |
| LinkQuest UWM4000 | 7 | 179.251 | 17 | 8.50 |
| Aquatech AQUAModem 1000 | 20 | 183.8103 | 9.75 | 2 |
| DSPComm AquaComm | 1.8 | 173.3527 | 23 | 0.48 |

Table 3-1: Commercial underwater modems and their specifications

### 3.1.3 Transmission Losses: Thorp Propagation model

A distinguishing property of acoustic channels is the fact that path loss depends on the signal frequency. This dependence is a consequence of absorption (i.e., transfer of acoustic energy into heat) [2]. The simplest model for acoustic attenuation in water is

$$A(d, f) = d^k \alpha\ (f)^d$$

Where $\alpha$: absorption coefficient factor, depends on the sound frequency $f$, $d$: Distance, and $k$: the spreading coefficient defined by geometry. The spreading factor, $k$, describes the geometry of propagation and is typically $1 \leq \alpha \leq 2$, e.g., $k = 1$, and $k = 2$ correspond to cylindrical and spherical spreading, respectively. $k = 1.5$ is often considered a practical setting. A cylindrical spreading corresponds to cases in which the transmission distance l is much larger than the depth of the ocean. In this case, the ocean bottom and the interface between the ocean and the air act as boundaries for the spreading of acoustic waves. On the other hand, spherical spreading is considered when the transmission distance is smaller than the depth of the ocean. This type of spreading provides a similar $k$ as the free-space approximation for radio wireless communications [3].

A common empirical formula used for absorption $\alpha(f)$ is Thorp's formula, which for $f$ in kHz is given by

$$10\, log_{10}\, \alpha(f) = \begin{cases} 0.11 \times \left(\frac{f^2}{1+f^2}\right) + \ +44 \times \left(\frac{f^2}{4100+f^2}\right) + 2.75 \times 10^{-4} \times f^2 + 0.003 & f \geq 0.4 \\ 0.02 + 0.11 \times \left(\frac{f}{1+f}\right) + 0.011 \times f & f < 0.4 \end{cases}$$

This output is in $db/km$. Combining absorption effects and spreading loss, the total attenuation is as follows:

$$10\ log\ A(d,f) = \ k \times \ 10\ log\ (d_m)\ + d_{km} \times 10\ log\ \alpha(f)$$

Here $d_m$ is in distance in meters, $d_{km} = \frac{d_m}{1000}$, and $f$ is in $KHz$. The Throp model outputs the attenuation in dB.

### 3.1.4 Noise

The calculation for the ambient noise in the underwater environment is divided into the major factors contributing to the total: turbulence, shipping, wind, and thermal. The following formulae give the noise power$[dB]$ of the four components

$$10\ logN_t(f) = \ 17\ -\ 30 log\ (\,f)$$

$$10\ logN_s(f) = \ 40\ +\ 20 \times (s\ -\ 0.5) + \ 26\ log\ f\ - 60 \times log(f\ +\ 0.03)$$

$$10\ logN_w(f) = \ 50\ +\ 7.5 \times \sqrt{w}\ +\ 20\ log\ f - 40\ log(f\ +\ 0.4)$$

$$10\ logN_{th}(f)\ =\ -15\ +\ 20\ log\ f$$

Where $f$ is frequency in kHz, $s$ is the shipping factor and $w$ is the wind-speed in $m/s$. The default value for $s$ is 0.5 and for $w$ is 0.

Turbulence noise influences only the very low frequency region, $f < 10\ Hz$. Noise caused by distant shipping is dominant in the frequency region $10\ Hz\ - 100\ Hz$, and it is modeled through the shipping activity factor $s$, whose value ranges between 0 and 1 for low and high activity, respectively. Surface motion, caused by wind-driven waves (w is the wind speed in m/s) is the major factor contributing to the noise in the frequency region $100\ Hz\ -\ 100\ kHz$ (which is the operating region of most acoustic systems). Finally, thermal noise becomes dominant for f > 100 kHz [4].

The total noise in linear scale is given as

$$N_{total}^{linear} = N_t^{linear} + N_s^{linear} + N_w^{linear} + N_{th}^{linear}$$

and the total noise in $dB$ domain is $N_{dB}^{total}[dB] = 10\log_{10}(N_{total}^{linear})$

### 3.1.5 Passive Sonar equation

NetSim uses the passive sonar equation to compute the $SNR$ at the receiver. The passive sonar equation is written as

$$SNR = SL - TL - (NL - DI)$$

Where $SL$ is the source level, $TL$ is the transmission losses, $NL$ is the noise level and $DI$ is receiver directivity index. NetSim assumes $DI = 0$, and upon appropriate substitutions the equation turns out as

$$SNR\,[dB] = SL\ - 10\log_{10} A(d,f)\,[dB] - \ N_{dB}^{total}[dB]$$

On the RHS of the equation, the second term is $TL$ and the final term is the $NL$.

### 3.1.6 MCS, Bit error rate (BER) and Packet error rate

Unlike 802.11 or 5G protocols, there are no standards specifying the modulation and coding scheme (MCS) for different received signal powers. Hence MCS is a user settable input parameter. Bit-error-rate (BER) is computed from the received signal to noise ratio (SNR) and the modulation and coding scheme (MCS) set by the user. NetSim assumes that the noise variance is Gaussian and the BER is computed from the standard SNR-BER tables described in the *Propagation Models* technology library manual. Packet error rate (PER) is then calculated from BER based on packet length.

Users who wish to use their own (custom) SNR-BER tables, can modify the UWAN_Calculate_BER function in UWAN.c file under the UWAN project.

### 3.1.7 Data Rate

The data rate is set through a GUI parameter; it is not computed by NetSim during simulation. In NetSim, the modulation and coding scheme (MCS) is set by the user in the GUI. This MCS setting does not affect the data rate; it only impacts BER calculations. More information on MCS is provided in section 3.1.6. The implicit assumption is that the user sets the right combination of data rate and MCS.

### 3.1.8 Collisions, Interference and Packet Capture

Per conventional definition collisions occur when there is simultaneous packet transmission, and these collided packets are assumed failed or lost. In this characterization, when there is simultaneous packet transmission, collisions are assumed to occur whatever the spatial separation (distance) between nodes and whatever the nodes' transmit powers. Clearly this is an approximation aimed at simplification. In case of simultaneous packet transmissions, it is possible that one (or more) signals have sufficient strength to be decoded by the receiver. Therefore, packets involved a collision can sometimes be successfully received. This phenomenon is known as Packet Capture.

NetSim approximately models Packet Capture considering interference threshold ($I_{th}$) rather than SINR. If the sum of all interfering signals, at the receiver, is greater than $I_{th}$ then the packet being received (by the receiver) is marked as collided. Else, if the total interference power is less than $I_{th}$ the packet succeeds despite ongoing simultaneous transmission. To elaborate, we start by denoting Underwater device as $UWD$ and let $UWD_{tx}$ be transmitting a packet to $UWD_{rx}$, while $UWD_I^n$ be $n$ other $UWDs$ that are transmitting at the same time. These are marked with subscript $I$ since these transmissions would interfere with the $UWD_{tx}$ to $UWD_{rx}$ transmission. NetSim computes the received signal power at $UWD_{rx}$ from all the other $n$ transmitting $UWDs$. All these powers are summed up and is equal to $\Sigma_{j=1}^{n} P_r^j$ where $P_r^j$ is the received power of interfering $UWD_I^j$ at $UWD_{rx}$. This summed interfering power, $\Sigma_{j=1}^{n} P_r^j$, is compared against an inference threshold $I_{th}$ which is equal to the receive sensitivity (set to $-85dB$ by default in NetSim). If total interfering power is greater than the interference threshold, then the packet being transmitted from $UWD_{tx}$ to $UWD_{rx}$ is marked as collided. Should the total interfering power be less than $-85dB$, the packet being transmitted from $UWD_{tx}$ to $UWD_{rx}$ is successfully received at $UWD_{rx}$.

An exact SINR based packet capture model (rather than the approximate Interference threshold-based packet capture model) is under development and is expected by the next release.

## 3.2 MAC Layer

NetSim currently only supports slotted Aloha in the MAC layer.

The reason for not supporting a contention-based protocol is: the basic principle of carrier sensing multiple access — that a node should transmit only if it hears no ongoing transmissions — is compromised in an acoustic channel where the packets propagate slowly, and the fact that none are overheard does not mean that some are not present in the channel [2].

### 3.2.1 Slotted Aloha

NetSim UWAN stack runs slotted Aloha (s-Aloha) in the MAC layer. s-Aloha is decentralized medium access control protocol that does not perform carrier sensing. Data transmission by devices is synchronized to timeslots. All devices are assumed to be perfectly synchronized to one another and to the timeslots.

In NetSim's s-Aloha implementation, the transmitter sends the next packet only after the current packet has either been received or errored. The knowledge of a packets successful or erroneous reception (due to collision or channel error) is known by the receiver only after it receives the complete packet. In case of unicast applications, the total time taken for the reception of a packet after the transmission commences is equal to the transmission time $T_{tx}$ plus the propagation delay Δ. It is assumed in NetSim, that the transmitter knows packet status at the exact time that the receiver knows the packet status. Hence the transmitter gets to know if a packet is successful or errored only $(T_{tx} + \Delta)$ seconds after commencing transmission. If the packet is errored the transmitter will retransmit (depending on the retry count set by the user in the UI) the packet; if the packet is successful, the transmitter will send the next packet.

### 3.2.2 Slot Length

Due to the long propagation delays user should take care to set the slot size in the GUI. This is a global parameter applicable to all UWAN devices. As a starting step, estimate the transmission time, $\mathrm{T_{tx}}$ which would be

$$T_{tx}(\mu s) = \frac{\left(L_{pkt} + OH\right) \times 8}{PHYRate}$$

where $L_{pkt}$is the application layer packet size, $OH$ is the overheads of all layers which is equal to 28B, and $PHYRate$ is the data rate set in the PHY layer. Next, the propagation delay, Δ is computed as $\Delta = \frac{d}{c_{sound}}$, where $d$ is the distance between the transmitter and the receiver. Thus, the ideal slot length should be

$$L_{slot} = \frac{\left(L_{pkt} + OH\right) \times 8}{PHYRate} + \frac{d}{C_{sound}}$$

*Example*: Let us consider the default values of $L_{pkt} = 14B$ and $PHYRate = 20\ Kbps$ which leads to $T_{tx} = 16{,}800\ \mu s$. Then using $t = \frac{25}{10} = 2.5$, $z = 50$, and $S = 35$ - where t is one-tenth of the temperature of the water in degrees Celsius, z is the depth in meters and S is the salinity of the water - we get (per earlier section 3.1) $c_{sound} = 2799.33\ m/s$. When the transmitter receiver distance is $d = 2km$, the propagation delay, $\mathit{\Delta} = \frac{2 \times 10^3}{2799.33} = 714{,}456.4\ \mu s$. Substituting all these, we see that the ideal slot length (when $d = 2km$) would be

$$L_{slot} = T_{tx} + \Delta = 16{,}800 + 714{,}456.4 = 731{,}256.4\ \mu s = 0.73\ s$$

Considering a slot length of $731{,}256.4\ \mu s$, we see if one packet exactly fits one slot then the predicted saturation throughput would be

$$\theta_{sat} = \frac{(L_{pkt} \times 8)}{L_{slot}} = \frac{(14 \times 8)}{731256.4 \times 10^{-6}} = 153\ bps$$

***NOTE:***

- ***If different Tx-Rx pairs are at different distances, then users should set the slot length based on the largest Tx-Rx distance i.e., based on the largest propagation delay.***
- ***NetSim limitation of not modeling link-level (reverse) ACKs can be overcome by adding additional time in the slot length to account for ACK transmissions. This work around, however, does not account for ACK failures.***

### 3.2.3 Retry count and Back-off

An important MAC layer parameter is Retry Count. This is a user-configurable parameter. A transmitter running s-Aloha retries when packets received (at the receiver) are in error. Recall here the NetSim assumption that the transmitter knows the packet status at the receiver.

The maximum number of times the transmitter retries is equal to Retry Count. Once this limit is hit the transmitter drops the packet. For example, if the Retry Count is 3, and if a packet fails in 1st transmission and in the 1st, 2nd and 3rd retransmission, then that packet is dropped by the transmitter.

The transmitter backs-off before each retry. The back off time (in slots) is a random number chosen from $[0, CW - 1]$ where $CW$ is the contention window and is equal to is $2^n$, where $n$ is the current value of the Retry Count. Unlike the 802.11 protocols, in s-Aloha, the transmitter does not back off before the 1st transmission.

The description above pertains to the behavior of unicast packets. For broadcast applications, the procedure varies as follows:

- Prior to the first transmission, the transmitter implements a back-off. The duration of this back-off is randomized, selected from a range of 0 to 7 slots.
- Broadcast packets do not undergo re-transmissions.

## 3.3 IP Addressing, Routing, Queuing and Buffers

Addressing for UWAN devices is IP based. The IP addresses are automatically set by NetSim.

### 3.3.1 Multi hop communication

The fact that the acoustic bandwidth depends on the distance has important implications for the design of underwater networks. Specifically, it makes a strong case for multihopping, since dividing the total distance between a source and destination into multiple hops enables transmission at a higher bit rate over each (shorter) hop [4].

NetSim currently supports static routing for multi-hop communication. Ad hoc routing is not yet available. While, no routing configuration is required for single hop communication, static routing needs to be configured for multi-hop communication. Static routing configuration is explained in the *Internetworks* technology library manual, Section: Configuring Static Routing in NetSim.

#### 3.3.1.1 SDN based Routing

Software defined networking (SDN) can be enabled in the application layer of UWAN devices. Complete details of SDN GUI interface and SDN working is given in Software Defined Networks technology library manual.

This feature will be available in Component 5: Software Defined Networks is licensed along with Component 12: UWAN.

### 3.3.2 Queuing and Buffers

Queuing in UWAN devices is on a first-in-first-out (FIFO) basis. For example, if a UWAN device is acting as both a source of traffic and a relay, then source traffic and relay traffic would arrive at the device's (MAC) buffer. The traffic would be queued and served per FIFO working. UWAN devices in NetSim have infinite (MAC) buffers.

## 3.4 Underwater Applications (Network Traffic Generation)

Users can model various kinds of applications (that generate network traffic). These include CBR, File transfer, Video etc. Typical examples include of transfers from one underwater device to another include (i) transferring a file (ii) transmitting very low bit rate video (iii) generating a constant-bitrate (CBR) application for theoretical performance studies etc. Details of the different application models are provided in Section 6 of the NetSim user manual.

# 4 Featured Examples

The configuration files (scenario, settings and other related files) of the examples discussed in this section are not in NetSim's UI (under built in examples) due to the large number of configuration files present (700+). Users can download the files from NetSim's UWAN git-repository.

**https://github.com/NetSim-TETCOS/UWAN-Experiments-and-Measurements-v14.0/archive/refs/heads/main.zip**

1. Click on the link given and download UWAN Experiments
2. Extract the zip folder. The extracted project folder consists of a NetSim Experiments file (UWAN experiments and measurements.netsimexp). How to import the workspace is explained in section 4.9.2 in NetSim User Manual.

## 4.1 Throughput and delay variation with distance

In this example, we understand how UWAN throughput and delay varies as the distance between 1 transmitter and 1 receiver is varied. Even with *No pathloss* the throughput in UWAN varies with Tx-Rx distance which is not the case in terrestrial RF based transmissions. The two parameters that affect throughput and delay are the speed of sound and the slot length of s-Aloha. The speed of sound in water is given by the formula

$$c_{sound} = 1449.05 + 45.7t - 5.21 \times t^2 + 0.23 \times t^3 + (1.333 - 0.126t + 0.009 \times t^2)(S - 35) + 16.3 \times z + 0.18 \times z^2$$

where $t$ is one-tenth of the temperature of the water in degrees Celsius, $z$ is the depth in km and $S$ is the salinity of the water in ppt. Then using $t = \frac{25}{10} = 2.5$, $z = 50$, and $S = 35$ - where t is one-tenth of the temperature of the water in degrees Celsius, z is the depth in meters and S is the salinity of the water - we get $c_{sound} = 2799.33\ m/s$. When the transmitter receiver distance is $d = 2km$, the propagation delay, $\Delta = \frac{2\times10^3}{2799.33} = 714{,}456.4\ \mu s$

Next, as explained in section 3.2.2, we consider ideal slot lengths for different transmitter receiver distances. In the case when $d_{Rx}^{Tx} = 2\ km$ the slot length turns out as

$$L_{Slot} = T_{tx} + \Delta = 16{,}800 + 714{,}456.4 = 731{,}256.4\ \mu s = 0.73$$

Table 4-4 shows the ideal slot length settings for $d_{Rx}^{Tx} = 4\ km$ and $d_{Rx}^{Tx} = 6\ km.$

**Network setup:**

- Create a scenario with 2 Underwater Devices

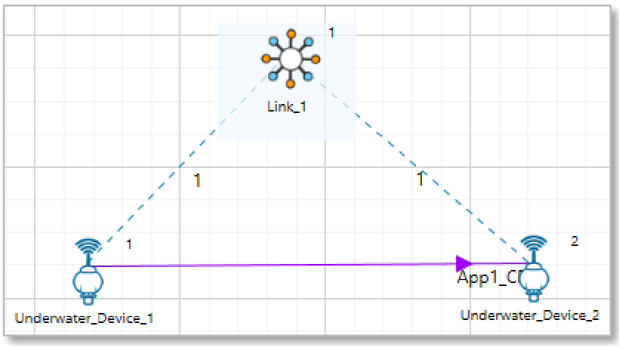



Figure 4-1: Network Scenario. Two underwater devices connected via an acoustic ad hoc link

- In case #1, distance between the underwater devices is set to be 2km. In case #2 the distance is 4km, while in case #3 it is set to 6 km
- Channel characteristics as NO_PATHLOSS

**Device Configuration:**

| **Device > Interface (ACOUSTIC) > Datalink Layer** | |
|---|---|
| **Slot Length(μs)** | 16800 |
| **Device> Interface (ACOUSTIC) > Physical Layer** | |
| **Source Level ($dB//1\mu Pa$)** | 200 |
| **Modulation** | QPSK |
| **Data Rate (kbps)** | 20 |

Table 4-1: Device properties set for this example

**Application Configuration:**

We run simulations for different traffic generation rates. The generation rate depends on the inter-packet arrival time – a GUI input in NetSim – in the following way

$$Generation\ Rate\ (Mbps) = \frac{Packet\ Size\ (Bytes)\ \times\ 8}{Interarrival\ Time\ (\mu s)}$$

| **Application Properties** | | |
|---|---|---|
| **Application Method** | App1_CBR | |
| **Source ID** | 1 | |
| **Destination ID** | 2 | |
| **Packet Size (Bytes)** | 14 | |
| | **Inter arrival Time (μs)** | **Generation rate (bps)** |
| **Case-1** | 4480000 | 25 |
| | 2240000 | 50 |
| | 1120000 | 100 |
| | 896000 | 125 |
| | 746666.6666 | 150 |
| | 640000 | 175 |
| | 560000 | 200 |
| **Case-2** | 5600000 | 20 |
| | 2800000 | 40 |

| | | |
|---|---|---|
| | 1866666.6666 | 60 |
| | 1400000 | 80 |
| | 1120000 | 100 |
| **Case-3** | 5600000 | 20 |
| | 3733333.333 | 30 |
| | 2800000 | 40 |
| | 2240000 | 50 |
| | 1866666.6666 | 60 |
| | 1600000 | 70 |

Table 4-2: Application properties for the different samples in each case studied in this example

- Click on Packet Trace option and select the Enable Packet Trace check box.
- Click on Logs option and Enable Acoustic Measurements Log.
- Run the Simulation for 100 sec.

**Theoretical Predictions**

The predicted propagation delay when the speed of sound $c_{sound} = 2799.33\ m/s$ is

| Distance between devices | Propagation delay ($\Delta$ in $\mu s$) |
|---|---|
| **2km** | 714456.4 |
| **4km** | 1428912.7 |
| **6km** | 2143369.1 |

Table 4-3: Theoretically predicted propagation delay for different Tx-Rx distances

**Transmission delay and Saturation Throughput**

Considering a slot length of $731{,}256.4\ \mu s$, we see that one packet exactly fits one slot and hence the predicted saturation throughput would be

$$\theta_{sat}^{2km} = \frac{(L_{pkt} \times 8)}{L_{slot}} = \frac{(14 \times 8)}{731256.4 \times 10^{-6}} = 153\ bps$$

Proceeding similarly for 4 km and 6 km, the predictions for saturation throughput are

| Distance between devices | Slot Length ($L_{slot}$) | Saturation Throughput ($\theta_{sat}$ in bps) |
|---|---|---|
| **2 km** | 731256.4 | 153 |
| **4 km** | 1445712.7 | 77 |
| **6 km** | 2160169.1 | 52 |

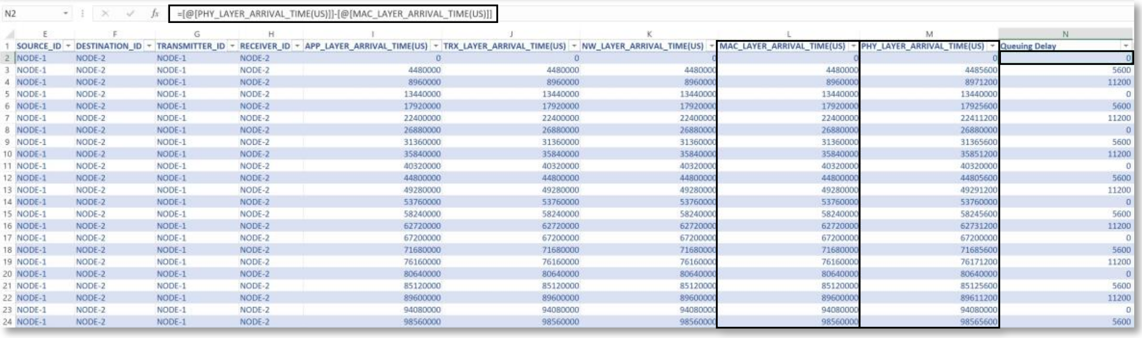
Table 4-4: Ideal slot lengths and theoretically predicted saturation throughputs ($\theta_{sat}$) for different Tx-Rx distances

**Simulation results**

We calculate of queuing delay, transmission delay, propagation delay from the packet trace. The steps are:

- Open Packet Trace file using the Open Packet Trace option available in the Simulation Results window.
- The difference between the PHY LAYER ARRIVAL TIME(US) and the MAC LAYER ARRIVAL TIME(US) will give us the delay of a packet. (Refer Figure 4-2)

$$Queuing\ Delay\ (\mu s) = PHYSICAL\ LAYER\ ARRIVAL\ TIME(\mu s) - MAC\ LAYER\ ARRIVAL\ TIME\ (\mu s)$$



N2 =[@[PHY_LAYER_ARRIVAL_TIME(US)]]-[@[MAC_LAYER_ARRIVAL_TIME(US)]]

| | E | F | G | H | I | J | K | L | M | N |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | SOURCE_ID | DESTINATION_ID | TRANSMITTER_ID | RECEIVER_ID | APP_LAYER_ARRIVAL_TIME(US) | TRX_LAYER_ARRIVAL_TIME(US) | NW_LAYER_ARRIVAL_TIME(US) | MAC_LAYER_ARRIVAL_TIME(US) | PHY_LAYER_ARRIVAL_TIME(US) | Queuing Delay |
| 2 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 4480000 | 4480000 | 4480000 | 4480000 | 4485600 | 5600 |
| 4 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 8960000 | 8960000 | 8960000 | 8960000 | 8971200 | 11200 |
| 5 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 13440000 | 13440000 | 13440000 | 13440000 | 13440000 | 0 |
| 6 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 17920000 | 17920000 | 17920000 | 17920000 | 17925600 | 5600 |
| 7 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 22400000 | 22400000 | 22400000 | 22400000 | 22411200 | 11200 |
| 8 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 26880000 | 26880000 | 26880000 | 26880000 | 26880000 | 0 |
| 9 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 31360000 | 31360000 | 31360000 | 31360000 | 31365600 | 5600 |
| 10 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 35840000 | 35840000 | 35840000 | 35840000 | 35851200 | 11200 |
| 11 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 40320000 | 40320000 | 40320000 | 40320000 | 40320000 | 0 |
| 12 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 44800000 | 44800000 | 44800000 | 44800000 | 44805600 | 5600 |
| 13 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 49280000 | 49280000 | 49280000 | 49280000 | 49291200 | 11200 |
| 14 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 53760000 | 53760000 | 53760000 | 53760000 | 53760000 | 0 |
| 15 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 58240000 | 58240000 | 58240000 | 58240000 | 58245600 | 5600 |
| 16 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 62720000 | 62720000 | 62720000 | 62720000 | 62731200 | 11200 |
| 17 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 67200000 | 67200000 | 67200000 | 67200000 | 67200000 | 0 |
| 18 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 71680000 | 71680000 | 71680000 | 71680000 | 71685600 | 5600 |
| 19 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 76160000 | 76160000 | 76160000 | 76160000 | 76171200 | 11200 |
| 20 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 80640000 | 80640000 | 80640000 | 80640000 | 80640000 | 0 |
| 21 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 85120000 | 85120000 | 85120000 | 85120000 | 85125600 | 5600 |
| 22 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 89600000 | 89600000 | 89600000 | 89600000 | 89611200 | 11200 |
| 23 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 94080000 | 94080000 | 94080000 | 94080000 | 94080000 | 0 |
| 24 | NODE-1 | NODE-2 | NODE-1 | NODE-2 | 98560000 | 98560000 | 98560000 | 98560000 | 98565600 | 5600 |

Figure 4-2 : Screen shot of NetSim trace showing the Queuing Delay column

- Now, calculate the mean queuing delay by taking the average of the queueing delays of all the packets. This is nothing but the column average. (Refer Figure 4-2)
- Similarly, users can get the Mean Transmission Delay and Mean Propagation Delay from the packet trace using the formulas

$$Transmission\ Delay\ (\mu s) = PHY\ LAYER\ START\ TIME(\mu s) - PHY\ LAYER\ ARRIVAL\ TIME(\mu s)$$

$$Propagation\ Delay\ (\mu s) = PHY\ LAYER\ END\ TIME(\mu s) - PHY\ LAYER\ START\ TIME(\mu s)$$

| | Generation Rate (bps) | Throughput (bps) | Delay (µs) | Mean Propagation Delay(µs) | Mean Transmission Delay(µs) | Mean Queuing Delay(µs) |
|---|---|---|---|---|---|---|
| Case #1: Distance between underwater devices is 2km | 25 | 26 | 736612.9 | 714456.4 | 16800 | 5356.52 |
| | 50 | 50 | 736856.4 | 714456.34 | 16800 | 5600 |
| | 100 | 100 | 736793.4 | 714456.4 | 16800 | 5537.08 |
| | 125 | 124 | 736856.4 | 714456.4 | 16800 | 5600 |
| | 150 | 149 | 738666.9 | 714456.4 | 16800 | 7410.53 |
| | 175 | 151 | 7377656.4 | 714456.4 | 16800 | 6646400 |
| | 200 | 151 | 12737656.4 | 714456.4 | 16800 | 12006400 |
| Case #2: Distance between underwater devices is 4km | 20 | 20 | 1451313 | 1428912.7 | 16800 | 5600 |
| | 40 | 40 | 1451313 | 1428912.7 | 16800 | 5600 |
| | 60 | 59 | 1453285 | 1428912.7 | 16800 | 7572.33 |
| | 80 | 76 | 3509313 | 1428912.7 | 16800 | 2063600 |
| | 100 | 76 | 12889313 | 1428912.7 | 16800 | 11443600 |
| Case #3: Distance between underwater devices is 6km | 20 | 20 | 2165769 | 2143369.1 | 16800 | 5600 |
| | 30 | 30 | 2167636 | 2143369.1 | 16800 | 7466.67 |
| | 40 | 39 | 2165609 | 2143369.1 | 16800 | 5440 |
| | 50 | 49 | 2165769 | 2143369.1 | 16800 | 5600 |
| | 60 | 52 | 8922169 | 2143369.1 | 16800 | 6762000 |
| | 70 | 52 | 14922169 | 2143369.1 | 16800 | 12762000 |

Table 4-5: Tabulated results (throughput and delays) for 3 different Tx-Rx distances

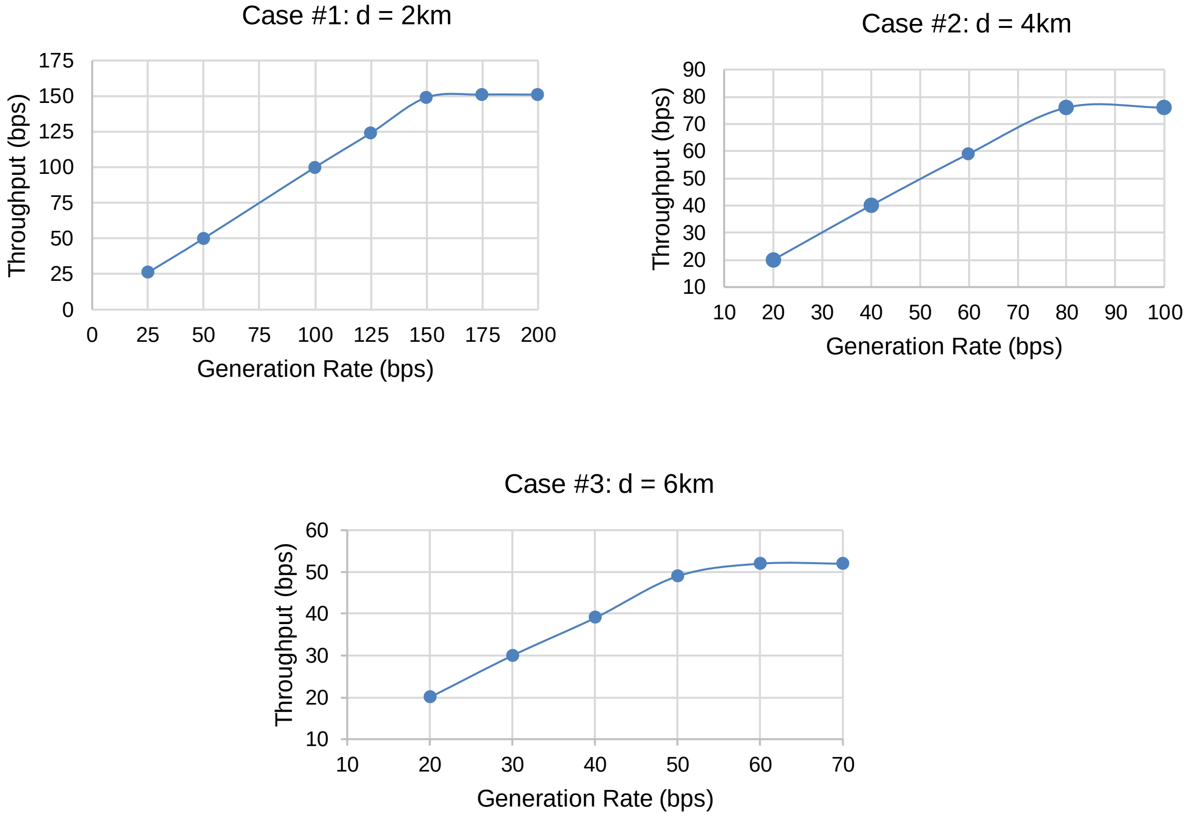



Figure 4-3: Throughput vs. Generation rate plotted for Tx-Rx distances of 2km, 4km and 6km based on earlier tables

From Table 4-5, we see that the propagation delays from simulation match predictions in Table 4-3. Then from we observe that saturation throughput (the Y axis value once the curve flattens) matches prediction.

| Distance between devices | Saturation Throughput. Predicted ($\theta_{sat}$ in bps) | Saturation Throughput. Simulation ($\theta_{sat}$ in bps) |
|---|---|---|
| **2 km** | 153 | 151 |
| **4 km** | 77 | 76 |
| **6 km** | 52 | 52 |

Table 4-6: NetSim UWAN Simulation results vs. theoretical prediction of saturation throughput, for different Tx-Rx distances

## 4.2 Underwater propagation losses and device range

In this example, we understand the Thorp propagation model, the sources of under-water noise, the passive sonar equation and how device range can be estimated based on received SNR. Refer to section 3.1 for the underlying theory on signal level, transmission losses, and the passive sonar equation.

## Network setup

- Create a scenario with two underwater devices

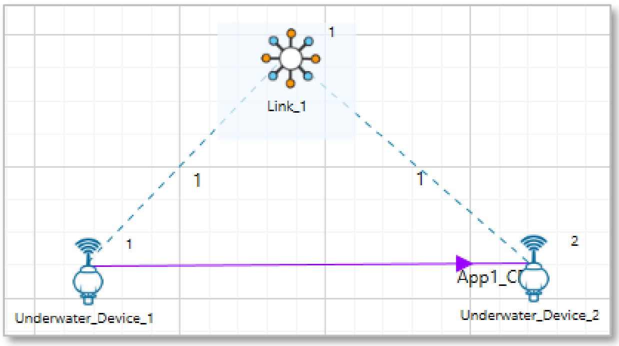



Figure 4-4: Network Scenario

- Channel characteristics as PATHLOSS_ONLY
- Change the Following parameters, Right Click on Underwater_Device_1 →Properties

| Device Properties > Physical Layer | |
|---|---|
| Source Level ($\boldsymbol{dB//1\mu Pa}$) | 190.8, 187.78,183.91 |
| Data Rate (kbps) | 20 |
| Modulation Technique | QPSK, BPSK, FSK, 16QAM, 64QAM, 256QAM |

Table 4-7: Device Properties

- Create a CBR Application with Default Properties with Source ID as 1 and Destination ID as 2.
- Click on Logs option and Enable Acoustic Measurements Log.
- Run the Simulation for 1000 sec.

## Analytical computations

In the Thorp model, the $db/km$ attenuation is given by.

$$10\, log_{10}\, \alpha(f) = \begin{cases} 0.11 \times \left(\dfrac{f^2}{1+f^2}\right) + \ +44 \times \left(\dfrac{f^2}{4100+f^2}\right) + 2.75 \times 10^{-4} \times f^2 + 0.003 & f \geq 0.4\, kHz \\ 0.02 + 0.11 \times \left(\dfrac{f}{1+f}\right) + 0.011 \times f & f < 0.4\, kHz \end{cases}$$

For this example, substituting $f = 20$, we get $10\, log_{10}\, \alpha(f) = 4.133\, db/km$, we see that the total pathloss is

$$10\, log\, A(d, f) = \ k \times \ 10\, log\, (d_m) \ + d_{km} \times 10\, log\, \alpha(f)$$

Using input parameters $K\ (spread\ coefficient) = 2, f\ = 20\ kHz$ and distance between the source and destination, $d\ = 18\ km$, and we get the total transmission loss, $TL$, as

$$TL = 10\ log\ A(d, f) = 153.51\ dB$$

Next, we turn to noise level $NL$. The turbulence, shipping, wind, and thermal, noise level in dB is given by

$$10\ logN_t(f) = 17 - 30log\ (f)$$

$$10\ logN_s(f) = 40 + 20 \times (s - 0.5) + 26\ log\ f - 60 \times log(f + 0.03)$$

$$10\ logN_w(f) = 50 + 7.5 \times \sqrt{w} + 20\ log\ f - 40\ log(f + 0.4)$$

$$10\ logN_{th}(f) = -15 + 20\ log\ f$$

Substituting $f = 20\ kHz$, shipping factor $s = 0.5$, surface windspeed $w = 0\ m/s$, we get $N_t = -22.03\ dB$, $N_s = -4.27\ dB$, $N_w = 23.63\ dB$, and $N_{th} = 11.02\ dB$. As explained in section 3.1.4 we see that wind noise has the most impact. After adding these noises in the linear scale and then converting back to $dB$, Total noise, $N_{Total}^{dB} = 23.87$. From the passive sonar equation

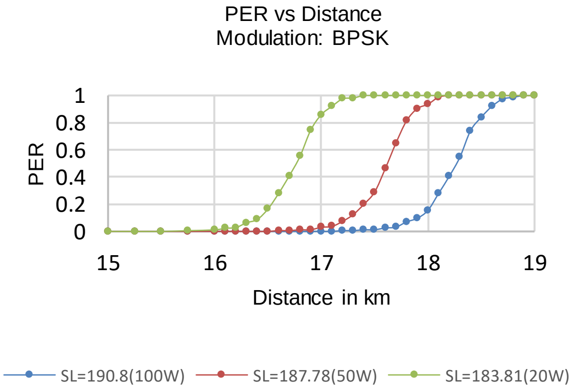
$$SNR = SL - TL - (NL - DI)$$

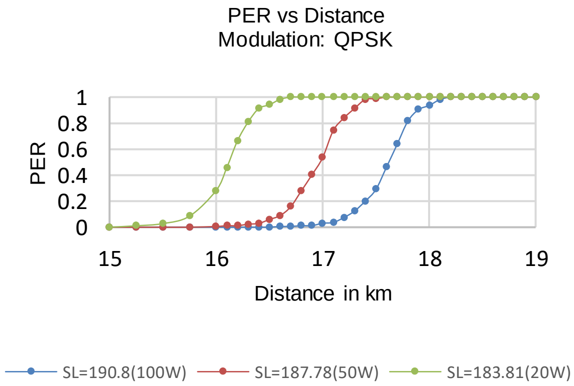
Substituting we get

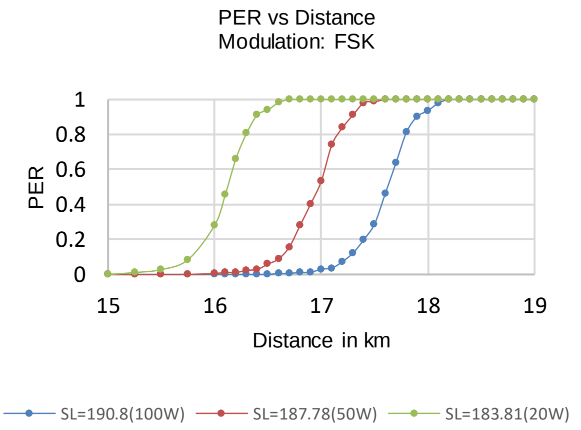
$$SNR = 190.80 - 153.51 - (23.87 - 0) = 7.41\ dB$$

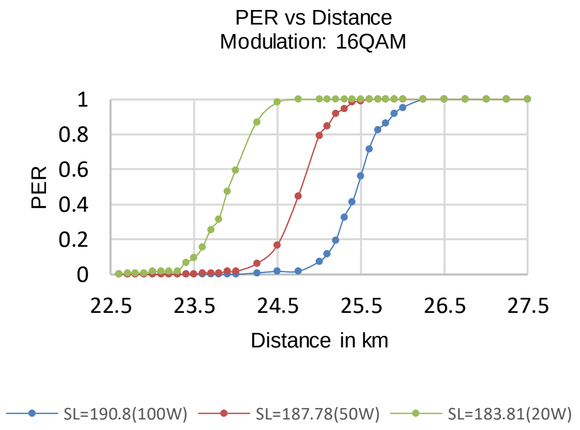
**Results: Packet Error Rate vs Distance**

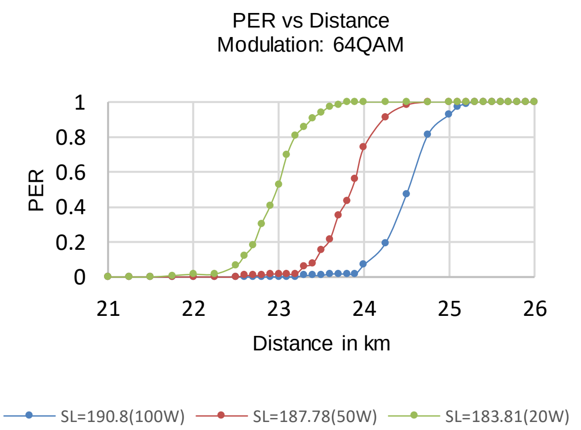
For the above SNR, we plot PER vs. distance for different modulation schemes given default packet size of 14B.

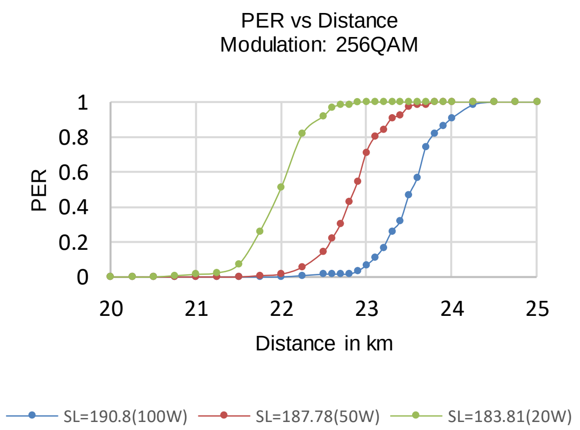





Generally, range is defined as the Tx-Rx distance at which the PER is 10 %. From these plots we can determine a device's range. In summary, we see how the device range is dependent on Source Level, Noise, MCS and packet size.

## 4.3 s-Aloha performance with multiple transmit nodes

### Network setup

We consider three scenarios as shown in the figure below, with 2, 3 and 4 transmitting nodes.

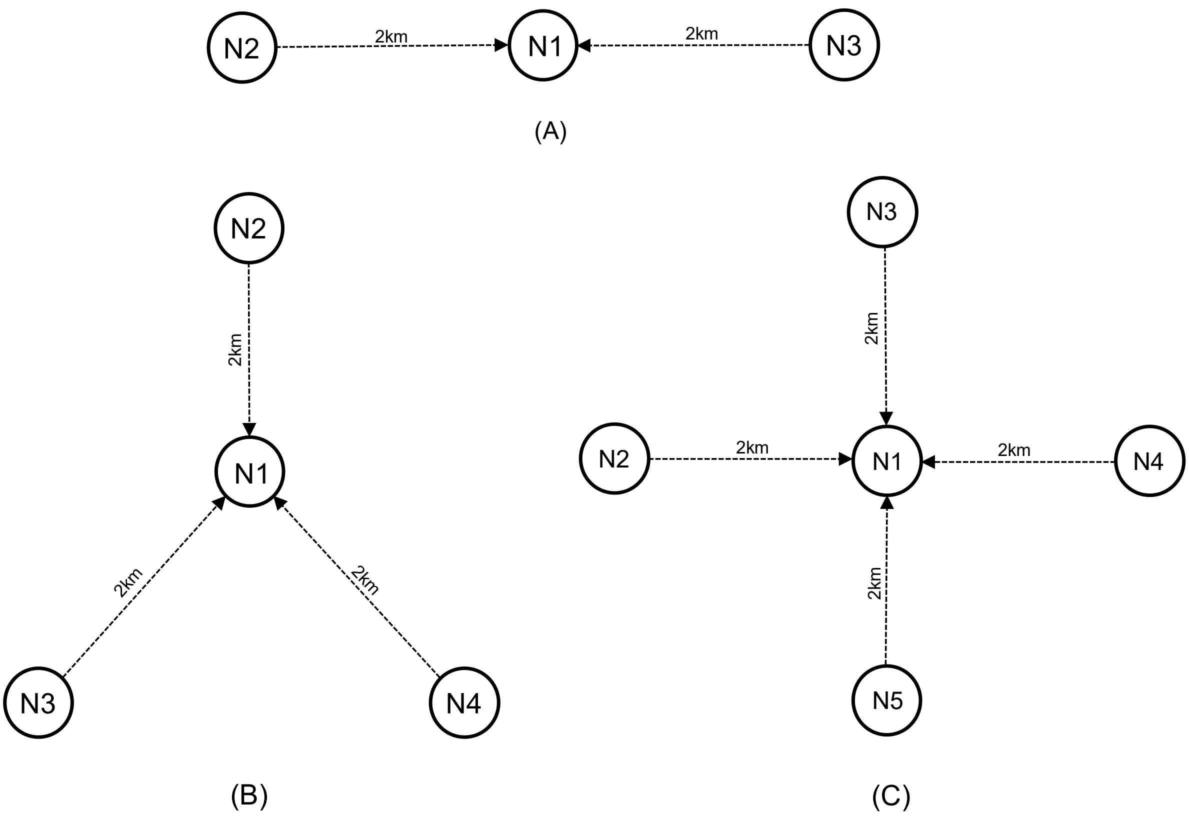



Figure 4-5: Simulation scenarios with 2 transmitting nodes in (A), 3 transmitting nodes in (B) and 4 transmitting nodes in (C). In all cases there is a single receiver.

## Properties

Then we set the UWAN device properties as shown below

| **Device Properties** | |
|---|---|
| **Device > Interface (ACOUSTIC) > Datalink Layer** | |
| **Retry Limit** | 0,4,6 |
| **Slot Length(μs)** | 741256.4 |
| **Device> Interface (ACOUSTIC) > Physical Layer** | |
| **Source Level ($dB//1\mu Pa$)** | 190.8 |
| **Modulation** | QPSK |
| **Data Rate (kbps)** | 20 |

Table 4-8: UWAN Device Properties

- Here, we set the Slot Time as 741256.4 $\mu s$, which is the ideal value of 731256.4 $\mu s$ plus a guard interval of 10,000 $\mu s$
- Create a CBR Application from the source nodes (2, 3, 4 and 5 per the cases) to the destination (Node 1) with a packet size of 14 bytes and Inter arrival time as 560000$\mu s$.
- Click on Logs option and Enable Acoustic Measurements Log.
- Run the Simulation for 10000sec.

## Results

We observe throughputs from network metrics and packets transmitted and packets collided from the Link Metrics. We collision probability as $P_c = \frac{Collision\ Count}{Packet\ Transmitted}$ and tabulate the results in the different cases.

| Case #1: Two transmitting nodes | | | | | | |
|---|---|---|---|---|---|---|
| **Retry Limit** | **Throughput N1 (bps)** | **Throughput N2 (bps)** | **Aggregate Throughput(bps)** | **Collision Count** | **Packet Transmitted** | $P_c$ |
| **0** | 0 | 0 | 0 | 26980 | 26980 | 1 |
| **4** | 55 | 51 | 106 | 7104 | 16624 | 0.427 |
| **6** | 71 | 65 | 136 | 2548 | 14647 | 0.173 |

Table 4-9: Simulation Results with 2 transmitting nodes

| Case #2: Three transmitting nodes | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Retry Limit** | **Throughput N1 (bps)** | **Throughput N2 (bps)** | **Throughput N3 (bps)** | **Aggregate Throughput (bps)** | **Collision Count** | **Packet Transmitted** | $P_c$ |
| **0** | 0 | 0 | 0 | 0 | 40470 | 40470 | 1 |
| **4** | 26 | 26 | 27 | 79 | 12234 | 19348 | 0.632 |
| **6** | 43 | 41 | 37 | 121 | 4969 | 15726 | 0.316 |

Table 4-10: Simulation Results with 3 transmitting nodes

| Case #3: Four transmitting nodes | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Retry Limit** | **Throughput N1 (bps)** | **Throughput N2 (bps)** | **Throughput N3 (bps)** | **Throughput N4 (bps)** | **Aggregate Throughput (bps)** | **Collision Count** | **Packet Transmitted** | $P_c$ |
| **0** | 0 | 0 | 0 | 0 | 0 | 53960 | 53960 | 1 |
| **4** | 15 | 15 | 14 | 16 | 60 | 16942 | 22293 | 0.75 |
| **6** | 28 | 27 | 26 | 26 | 107 | 7202 | 16756 | 0.42 |

Table 4-11: Simulation Results with 4 transmitting nodes

We carry out simulations with different settings of Retry Count. The final results are plotted below. When Retry count is set to zero, all packets collide even when just two nodes are

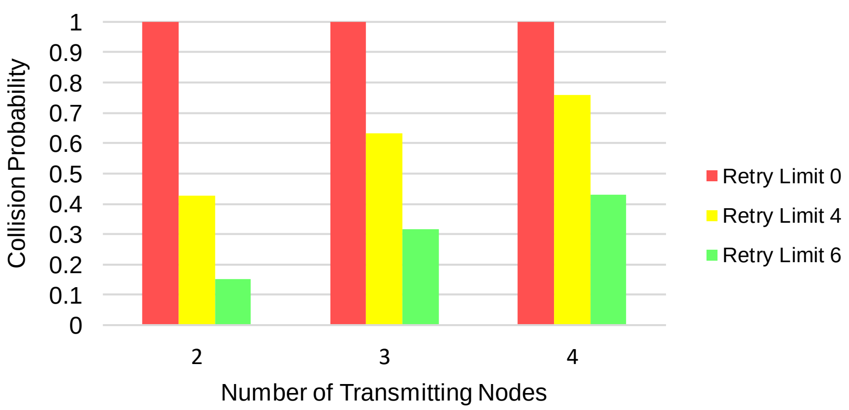



Figure 4-6: Collision probability vs Number of Transmitting Nodes

transmitting. With retry count set to 0, the node attempts a packet transmission. If it fails, there is no retry and the packet is dropped. Recall, that in s-Aloha the transmitter does not back off before the first transmission attempt for a packet. With backlogged queues, the two transmitting nodes keep attempting at each slot. This leads to continuous collisions.

When the retry count is set to 4 (or 6) a transmitting node back off per the exponential backoff algorithm, before every retransmission. The back off algorithm is explained in section 3.2.3. Hence there is an element of randomness in packet transmissions at each slot. Nodes may or may not transmit. The probability of transmission at a particular slot reduces as the Retry Count is increased. Hence, we see lower collision probabilities for Retry count of 6.

**Advanced:** s-Aloha works as expected only if the slot time is greater than transmission time plus the propagation delay. What happens when slot length is lower than this limit? Under this condition a new slot is available for transmission even before completion of an existing packet transmission. Thus, if an idle node completes its back off before the transmission (where transmission time equals $T_{tx} + \Delta$) of one another node is complete, it will attempt to transmit at the start of a new slot. The two packets collide and are lost. One would therefore expect higher collision probabilities with smaller slot lengths. NetSim simulation results for 2-node transmitting case with slot time, 250 $ms$ is given below

| Retry Limit | Throughput N1 (bps) | Throughput N2(bps) | Aggregate Throughput(bps) | Collision Count | Packet Transmitted | $P_c$ | $P_c$ (Ideal slot time) |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 26666 | 26666 | 1 | 1 |
| 4 | 20 | 20 | 40 | 14102 | 17696 | 0.797 | 0.427 |
| 6 | 45 | 46 | 91 | 6987 | 15109 | 0.462 | 0.153 |

Table 4-12: Simulation Results with slot time vs collision probability

Observe the significant increase in collision probability when the slot time is set lower than the ideal slot time. Next, we fix the retry count at 6 and vary the slot time from 50 $ms$ to 750 $ms$ in steps of 50 $ms$. In the plot below showing collision probability vs. slot time, we again observe how low slot times lead to higher collision probabilities.

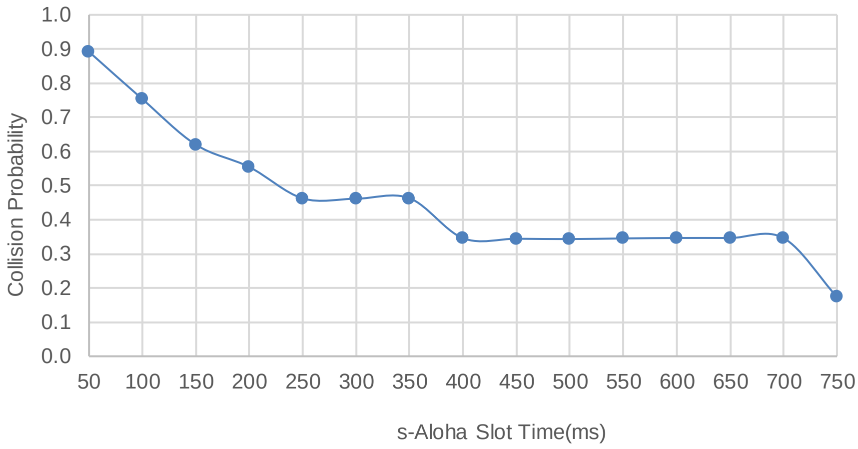



Figure 4-7: Collision probability vs slot time

# 5 Limitations

- Users cannot model buoy nodes, behaving as network sink(s) to collect data which can then be transmitted, via Radio Frequency (RF), to a nearby boat, or satellite
- Link layer ACKs are not modeled in s-Aloha
- There is no Guard time between slots in s-Aloha
- Dynamic routing protocols are not supported
- Doppler shift on account of transmitter and/or receiver motion is not accounted
- UWAN cannot connect to an “external” network; it operates stand alone

# 6 References

[1] H. U. Yidliz, V. C. Gungor and B. Talvi, "Packet Size Optimization for Lifetime Maximization in Underwater Acoustic Sensor Networks," IEEE Transactions on Industrial Informatics, 2019.

[2] M. Stojanovic and J. Preisig, "Underwater Acoustic Communication Channels: Propagation Models and Statistical Characterization," IEEE Communications Magazine, 2009.

[3] D. E. Lucani, M. Medard and M. Stojanovic, "Capacity Scaling Laws for Underwater Networks," 2012.

[4] M. Stojanovic, "On the Relationship Between Capacity and Distance in an Underwater Acoustic Communication Channel," in WUWNet, 2006.

[5] R. J. Urick, Principles of Underwater Sound, Peninsula Publishing, 1983.