Abstract—We propose a location-free link state routing protocol for Underwater Acoustic Sensor Networks (UASNs). Additionally, we present the mathematical background for the theoretical capacity and transmission power metrics of an underwater acoustic channel. UASNs are formed by devices enabled with acoustic communication capabilities that are deployed underwater to perform collaborative monitoring tasks. Information is collected by a sink at the surface also equipped with a radio. The underwater communication channel is characterized by a limited bandwidth and high propagation delay. The network topology constantly changes due to mobility of the nodes. In our routing protocol, every node ranks the quality of the path that it offers toward the sink. Packet forwarding is performed hop-by-hop considering one or several routing metrics, e.g., hop count or pressure. To avoid communication void problems, every node selects a one-hop neighbor within an area that guarantees progress toward a sink. Our strategy is loop-free. It includes a recovery mode handling network topology changes. Our routing protocol was implemented in ns-3 to conduct experiments.

I. INTRODUCTION

Underwater Acoustic Sensor Networks (UASNs) consist of devices enabled with acoustic communication capabilities that are deployed underwater at different depths to perform collaborative monitoring tasks [1]. Underwater nodes gather and send information to a sink that is also equipped with a radio to communicate with other network components located on the surface. We present a location-free routing protocol designed for UASNs. Research in UASNs focuses on both physical and link layers [2], [3], whereas research on the network layer is still in an early stage. The design of an efficient routing mechanism should consider the limitations of the medium. The underwater acoustic channel is characterized by a high bit error rate, low data rate and large propagation delay. Underwater routing protocols must be energy-aware since the deployment and maintenance of underwater devices are particularly difficult. UASNs are formed by nodes in constant motion that leads to continuous changes in the network topology.

Electromagnetic and optical waves do not work well underwater due to the nature of the medium, especially in the case of seawater. Acoustic waves are used for underwater communication due to the relatively low attenuation (i.e., signal reduction) of sound in water, specially in thermally stable, deep water settings. In shallow water, acoustic waves are severely affected by temperature, site specific noise and multipath propagation due to reflection and refraction. The speed of sound in water varies according to the depth and is affected by temperature, salinity and pressure. The speed of acoustic waves is about 1500 m/s [4] close to the ocean surface, which is more than four times faster the speed of sound in air, but five orders of magnitude smaller than the speed of light [5]. Compared with electromagnetic and optical waves in terrestrial networks, the speed of acoustic waves is significantly lower. As a consequence, underwater channel communication is also affected by a severe Doppler effect.

The characteristics of the underwater acoustic communication channel require new efficient and reliable data communication protocols. Underwater communication is characterized by a path loss (i.e., path attenuation) that depends not only on the distance between the transmitter and receiver, but also on the signal frequency [2]. Low frequency waves (e.g., below 10 kHz) are effective for long-range communication [5]. Bandwidth for underwater communication is typically in the order of kHz. Routing protocols for ad hoc networks are not suitable for underwater communication. Proactive routing protocols require a constant exchange of control information to keep the routing information up to date. In reactive approaches, the route discovery process is affected by an increased delay.

Several routing protocols in UASNs are based on a greedy hop-by-hop method for packet delivery [6]. Unlike the end-to-end routing, greedy hop-by-hop routing approaches select as next hop those one-hop neighbors that have positive progress toward the sink. Greedy routing protocols do not guarantee to find a path toward the sink, e.g., data packets reach a node with no positive progress. This problem is known as communication void. Routing protocols for UASNs can be classified as location-based or location-free. Location-based approaches assume that nodes know both their geographical position and the sink position. As a main drawback, finding the location information of nodes is a main challenge due to the inapplicability of GPS under the water. Location-free approaches can be divided in the pressure-based and beacon-based categories. In pressure-based routing protocols, the depth information...
(i.e., pressure) is used to identify the positive forwarding area. Beacon-based approaches implement beacon messages with information to reach the sink, e.g., distance in hops. Greedy routing protocols in UASNs do not consider the quality of the links. In our location-free link state routing approach, every node ranks the quality of the path that it offers toward a sink. Packet forwarding is performed hop-by-hop. Every node selects a one-hop neighbor with the lowest hop count toward a sink. If there is a tie, then the one-hop neighbor with the best quality path toward a sink is selected. If the tie persists, then the neighbor with lowest pressure is selected. Our strategy is loop-free. It includes a recovery mode handling topology changes. Our location-free routing protocol was implemented in ns-3. The UAN model developed by L. Tracy was adapted to interact with our routing protocol.

Organization of the paper — Section II describes the related work. Section III describes an example of the routing metric calculation in UASNs. Our location-free routing protocol is presented in Section IV. Section V presents our experiments and results. Section VI concludes the paper.

II. RELATED WORK

In this section we present greedy routing protocols designed for UASNs. We divide this section into location-based and location-free routing protocols.

A. Location-based routing in UASNs

In the location-based category it is assumed that nodes know their geographical position. Nodes also know the location of the sink. Using this information, a data packet sender establishes the conditions to reach the forwarders with positive progress toward the sink. The sender computes an area, i.e., a cone [7], pipeline [8] or a zone, to select the forwarders. Receivers within this area retransmit the data packet until it reaches the sink. The size of the forwarding area has a direct impact on the performance of the network. If the size is too big, then more nodes retransmit the packets increasing the network overhead and the overall energy consumption. If the size is too small, then the probability of having a communication void problem increases.

The Vector-Based Forwarding (VBF) [8] proposed by Xie et al., is a greedy location-based routing protocol designed for UASNs. Every node knows its own location and the sink location. Originators of packets create an imaginary vector to the sink with a predefined radius, e.g. node A. Information about the vector is included in the packets. Receivers use the radius information as a threshold. If their distance to the vector is less than the radius, then they are allowed to retransmit the packet. The protocol was designed to address the constant movement of nodes and energy consumption. VBF does not consider the communication void problem. The Hop-by-Hop VBF (HH-VBF) routing protocol is an enhanced version of VBF proposed by Nicolaou et al. in [9]. The protocol attempts to reduce the probability of communication void problem in sparse networks. In HH-VBF every forwarder computes its own pipeline toward the sink. This increases the chances of finding nodes with positive progress to the sink. The Focused Beam Routing (FBR), protocol is a location-based routing protocol. FBR uses a finite number of predefined power levels, i.e., $P_1, P_2, \ldots, P_n$, where $P_n$ is the highest power level. A sender of a data packet computes a cone with angle $\theta$ toward the sink. The radius size of the cone is determined by the power level, i.e., $P_i$. A request to send (RTS) message is generated by the sender. Receivers within the cone replay with a clear to send (CTS) message including its ID and location. The sender selects the closest node to the sink as a forwarder. Every node repeats the same procedure until the packet reaches the sink. If the sender does not receive any CTS message using a power level $P_i$, then it increases the power level, i.e., $P_{i+1}$, until a neighbor node is found or the power level is $P_n$. If there is no neighbor node inside the cone computed with power level $P_n$, then the cone can be rotated until finding a neighbor node. The performance of FBR relies on the selected angle $\theta$. The angle should be wide enough in sparse networks in order to avoid a communication void. In dense networks, the angle $\theta$ should be small enough to reduce overhead and energy consumption.

B. Location-free routing in UASNs

In the location-free category, nodes are not fully aware of their geographical position. Instead, they use other information such as depth or hop-count to route a data packet to the sink. This category can be divided in two subcategories: beacon-based and pressure-based. In the beacon-based subcategory, every node periodically broadcasts beacon messages in order to provide dynamic information for identifying positive progress toward the sink. In the pressure-based subcategory, only depth information measured locally by pressure sensors is used to identify a positive progress area toward the sink.

Depth-based routing (DBR) [10] is the first location-free greedy hop-by-hop routing protocol designed for UASNs. DBR assumes multiple sinks and nodes equipped with a pressure sensor to calculate their depth. The routing approach is as follows. Every sender broadcasts a data packet including a packet sequence number and its depth information. One-hop neighbors with lower depth value are candidates to forward the packet toward the sink. Receivers compute a holding time based on its depth before retransmitting. In order to reduce overhead caused by redundant packet transmission, every forwarder computes a different holding time. The closest node to the surface retransmits the packet first. Receivers reject a data packet either if it arrives from a node with lower depth or if it has been processed. DBR handles the constant movements of nodes but does not tackle the communication void problem.
III. CAPACITY AND POWER METRICS IN UASNs

We developed the mathematical background for the theoretical capacity and transmission power metrics of an underwater acoustic channel. Both capacity and power depend on attenuation and noise.

A. Attenuation and Noise

For an underwater acoustic channel, attenuation and noise are frequency dependent. For attenuation, we use the model of Thorp [11]. An underwater acoustic signal is attenuated during propagation. There are two main causes: conversion of acoustic energy into heat and geometrical spreading loss. Conversion of acoustic energy into heat is proportional to the frequency of the signal. Geometrical spreading component captures the fact that as sound wave travels away, the distance from the source increases. The surface covered by the waves expands. The sound energy intensity drops. These factors are represented in the Thorp model [2]:

\[ A(d, f)_{db} = k \cdot 10 \log d + d \cdot a(f) \]  

(1)

where \( d \) is the transmitter-receiver separation distance (in km), \( k \) is a geometrical spreading loss exponent (with values in the interval \([1, 2]\)), \( f \) is frequency (in kilo Hertz), and \( a(f) \) is a frequency-dependent absorption coefficient (in dB/km). The geometrical spreading loss exponent depends on the geometry of propagation, which can be cylindrical \((1)\), spherical \((2)\) or practical \((1.5)\). For a frequency \( f \) above a few hundred Hertz, the absorption coefficient can be determined using the equation:

\[ a(f)_{db/km} = \frac{0.11 \cdot f^2}{1 + f^2} + \frac{44 \cdot f^2}{4100 + f^2} + 2.75 \cdot 10^{-4} \cdot f^2 + 0.003 \]  

(2)

otherwise the following equation can be used:

\[ a(f)_{db/km} = 0.002 + \frac{0.11 \cdot f^2}{1 + f^2} + 0.011 \cdot f^2 \]  

(3)

There are two main kinds of noise affecting underwater acoustic communications: site-specific noise and ambient noise. Sources of site-specific noise are geographically dependent. They include noise made by breaking ice and sea creatures. Four sources of ambient noise have been identified: turbulence, shipping, waves and thermal. Ambient noise is approximated using the following equation [12]:\(^1\)

\[ N(f)_{dB \ re \ \mu \ Pa \ per \ Hz} = 50 - 18 \cdot \log f \]  

(4)

Alternatively, equations do exist modeling turbulence, shipping, waves and thermal noise, individually [12]. The resulting ambient noise is the sum of the evaluation of all of them. The three-dimensional surface of Fig. 1 shows the attenuation-noise (AN) product (in dB form) as a function of frequency, measured in k Hertz, and distance, in km. For each distance, there is a frequency where the attenuation-noise product reaches a minimum. This is the frequency where communication conditions are optimal for that distance.

\(^1\)The term \( \mu \ Pa \) stands for reference pressure 1 micropascal. The pascal is a unit of force per unit area, i.e., pressure. It is equal to one newton per square meter.

![Fig. 1: Attenuation-noise (AN) product, in dB re μ Pa, versus frequency, in Hertz, and distance, in km.](image)

B. Capacity and Power Calculation

There are two methods for estimating the bandwidth of an underwater acoustic channel [2]. The first is called 3 dB bandwidth. Among the available bandwidth, it finds the optimal frequency, i.e., with minimum attenuation-noise product. The Signal-to-Noise-Ratio (SNR) is determined for that frequency. The lower and upper sidebands are included, while the SNR stays below two times the minimum attenuation-noise product. This method has been implemented by Harris and Zorzi for NS2 [13]. The 3 dB bandwidth method does not guarantee optimality. Stojanovic describes a second method called optimal [2]. This method calculates the bandwidth according to a required \( SNR_0 \), e.g., 20 dB. As a function of frequency, the transmission power is distributed and adapted to the estimated attenuation and noise in the channel. Let \( A(d, f) \) denote the attenuation, in linear form, as a function of distance \( d \), in km, and frequency \( f \), in kilo Hertz. Let \( N(f) \) denote the noise, in \( \mu \ Pa \) per Hertz, as a function of frequency. Let \( P(f) \) denote the transmission power, in \( \mu \ Pa \) per Hertz, as a function of frequency. According to the water-filling principle [14], at distance \( d \), channel capacity is maximized (with respect to bandwidth) when the following equality holds:

\[ P(f) = K - A(d, f)N(f) \]  

(5)

In the sequel, \( K \) is called the water-filling constant. The optimal method determines the actual value of this constant. The optimal algorithm is as follows. Within the total channel bandwidth, the optimal frequency is identified. The optimal frequency \( f_{opt} \) is where the attenuation-noise product \( A(d, f)N(f) \) is minimal. The value of the water-filling constant \( K \) is initialized at a value slightly above the attenuation-noise product at the optimal frequency, i.e., \( K \) is set to the product \( 1.01 \cdot A(d, f_{opt})N(f_{opt}) \). The following procedure is executed repeatedly. Within the total available bandwidth, the sub-band, around frequency \( f_{opt} \) where the product \( A(d, f)N(f) \) is lower than or equal to \( K \) is determined. Let us assume that this sub-band extends from \( f_{min} \) to \( f_{max} \), with \( f_{min} \leq f_{opt} \leq f_{max} \). Assuming that the power is
distributed according to Equation 5, we have that
\[
SNR = \frac{\int_{f_{\text{min}}}^{f_{\text{max}}} P(f) A(d, f)^{-1} df}{\int_{f_{\text{min}}}^{f_{\text{max}}} N(f) df}.
\] (6)
Substituting \( K = A(d, f) N(f) \) for \( P(f) \) (Equation 5), we get
\[
SNR = K \cdot \frac{\int_{f_{\text{min}}}^{f_{\text{max}}} A(d, f)^{-1} df}{\int_{f_{\text{min}}}^{f_{\text{max}}} N(f) df} - 1.
\] (7)
If the \( SNR \) is lower than \( SNR_0 \), then the water-filling constant \( K \) is augmented using a multiplicative factor, e.g., the new \( K \) is \( 1.01 \cdot K \). When the \( SNR \) is greater than or equal to \( SNR_0 \), the procedure stops.

The channel capacity is
\[
C_{\text{bps}} = \int_{f_{\text{min}}}^{f_{\text{max}}} \log_2 \left( \frac{K}{A(d, f) N(f)} \right) df.
\] (8)
The transmission power is
\[
P_{\mu} P_a = \int_{f_{\text{min}}}^{f_{\text{max}}} P(f) df.
\] (9)
Substituting \( K = A(d, f) N(f) \) for \( P(f) \) (Equation 5), we get
\[
P_{\mu} P_a = K \cdot (f_{\text{max}} - f_{\text{min}}) - \int_{f_{\text{min}}}^{f_{\text{max}}} A(d, f) N(f) df.
\] (10)

IV. LOCATION-FREE ROUTING FOR UASNS

We present a new location-free UASN routing protocol. For the purposes of packet forwarding in the direction of the sink, every node selects a next hop according to link-state metrics. Figure 2 depicts an example. To prevent communication voids, the next hop is selected such that progress toward a sink, \( S \) in Figure 2, is guaranteed. It is assumed that nodes are equipped with pressure measurement sensors and have the same transmission range. Every node ranks the quality of the path that it offers toward the sink. Path quality is a metric that evaluates redundancy of routes. Every node generates beacon messages that include hop count, path quality toward a sink and pressure. Sinks generate beacon messages with path quality equal to one and hop count and pressure equal to zero. For an underwater node, the initial path quality is equal to zero. For an underwater node, the initial path quality is equal to one and hop count and pressure equal to zero. The steps to send a beacon message are illustrated in function SendBeacon. Upon receiving a beacon message, the receiver updates its path quality and hop count state. Function ReceiveBeacon illustrates the steps to process a beacon message. In Figure 2, arcs are labeled with beacon messages, parameterized with hop count (\( hc \)), path quality (\( pq \)) and pressure (\( pr \)). Beacon messages are tagged in the order they were generated. For instance, \( S \) starts generating beacon messages with hop count equal to zero, path quality equal to one and its pressure value, i.e., \( [0, 1, pr_S] \). After receiving the beacon message from sink \( S \), underwater nodes \( v_a \) and \( v_b \) generate a new beacon message with path quality and hop count equal to one. The one-hop neighbor with lowest hop count value is selected as next hop for packet forwarding. In case of a tie, the greatest path quality neighbor is selected. If the tie persists, then the lowest pressure neighbor is selected. Function SelectNextHop illustrates the steps to select the next hop. Initially, nodes \( v_a \) and \( v_b \) generate beacons with zero path quality and \( null \) hop count. After receiving the message from \( S \), both update their state. Node \( v_a \) generates beacon message \( [1, 1, pr_a] \). Node \( v_b \) produces \( [1, 1, pr_b] \). Node \( v_d \) receives beacon messages from both \( v_a \) and \( v_b \). The path quality of \( v_d \) becomes two, i.e., the sum of the path quality of neighbors \( v_a \) and \( v_b \). Assuming that it has the lowest pressure, node \( v_d \) selects \( v_a \) for packet forwarding. It is the neighbor closest to the surface. To reach the sink, \( v_g \) has two equal hop count options: \( v_d \) and \( v_e \). Node \( v_d \) is selected because it has the highest path quality. The routing protocol is loop-free and comprises a recovery mode handling network topology changes, e.g., when links are broken.

Fig. 2: Beacon messages include: hop count (\( hc \)), path quality (\( pq \)) and pressure (\( pr \)).

Fig. 3: Recovery Mode.
Get path quality of  
else 
   Increase hop count by one, 
Instance, suppose that in Figure 3, the link (v1, v2) is broken. For instance, some paths might become broken. Therefore, 
Nodes must be aware that nodes in our simulations. In every experiment, nodes are randomly distributed in a tree-dimensional space of 100m³. We simulate scenarios with five to eight nodes plus the sink. Every node has a transmission power of 160 dB re μ Pa and a minimum signal-to-interference-plus-noise-ratio (SINR) equal to 80dB. The maximum transmission range is 50 meters and the data rate is 80bps. We simulate every scenario during one thousand seconds (i.e., an event), after this time, we update the position of every node. The sink is static. We execute one hundred events per experiment. We use the end-to-end delay and number of data packets received by the sink as metrics to evaluate the performance of our routing protocol.

V. EXPERIMENTS AND RESULTS
We conducted our experiments using the ns-3 simulator [15], version 3.21 and the UAN model developed by L. Tracy. The UAN model was adapted to interact with our routing protocol. Table I describes the parameter we used in our simulations. In every experiment, nodes are randomly distributed in a tree-dimensional space of 100m³. We simulate scenarios with five to eight nodes plus the sink. Every node has a transmission power of 160 dB re μ Pa and a minimum signal-to-interference-plus-noise-ratio (SINR) equal to 80dB. The maximum transmission range is 50 meters and the data rate is 80bps. We simulate every scenario during one thousand seconds (i.e., an event), after this time, we update the position of every node. The sink is static. We execute one hundred events per experiment. We use the end-to-end delay and number of data packets received by the sink as metrics to evaluate the performance of our routing protocol.

A. Communication void and Recovery Mode
In UASNs the network topology changes very frequently. As a consequence, some paths might become broken. For instance, suppose that in Figure 3, the link (v1, v2) is broken. Nodes generate beacon messages according to the network topology changes. The first goal is to avoid communication void problems. Nodes must be aware that v1 has lost communication with the sink. Therefore, v1 generates beacon messages with a path quality value equal to zero and hop count equal to infinity. v1 accepts beacon messages from v2 because it has the lowest hop count value. v2 updates its hop count toward the sink and generates beacon messages with a path quality value equal to zero. Receivers update their local information about the network topology and select as next hop, nodes in an area without communication voids. For instance, v2 and v3 select v4 as the next hop to reach the sink. The second goal is to provide a mechanism to find new routes to reach a sink when the network topology changes, i.e., a recovery mode. For instance, v4 has two one-hop neighbors with the same hop count toward the sink, i.e., v5 and v6. v5 selects v6 as next hop because it has the highest positive degree value.

<table>
<thead>
<tr>
<th>Function SendBeacon(v)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Compute pressure, i.e., prv ← Pressure(v);</td>
</tr>
<tr>
<td>2 if (v is a sink) then</td>
</tr>
<tr>
<td>3 Hop count is equal to zero, hc ← 0;</td>
</tr>
<tr>
<td>4 Path quality is equal to one, pq ← 1;</td>
</tr>
<tr>
<td>5 else</td>
</tr>
<tr>
<td>6 Select next hop. NextHop ← SelectNextHop(v);</td>
</tr>
<tr>
<td>7 Get its hop count, hc ← HopCount(NextHop);</td>
</tr>
<tr>
<td>8 if (Hop count is equal to null) then</td>
</tr>
<tr>
<td>9 Path quality is equal to zero, pq ← 0;</td>
</tr>
<tr>
<td>10 else</td>
</tr>
<tr>
<td>11 Get path quality of v, pq ← PathQuality(v);</td>
</tr>
<tr>
<td>12 Increase hop count by one, hc ← hc + 1;</td>
</tr>
<tr>
<td>13 v generates a new beacon message, [hc, pq, prv];</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function SelectNextHop(v) → NextNode</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Initialize the set nNodes as empty set, nNodes ← ∅;</td>
</tr>
<tr>
<td>2 Select among the one-hop neighbors of v, N1(v), the lowest hop count neighbor, nNodes ← lowestHopCount(N1(v));</td>
</tr>
<tr>
<td>3 if (there is a tie,</td>
</tr>
<tr>
<td>4 Select the highest path quality neighbor, nNodes ← highestPathQuality(nNodes);</td>
</tr>
<tr>
<td>5 if (the tie persists,</td>
</tr>
<tr>
<td>6 Select the lowest pressure neighbor, nNodes ← lowestPressure(nNodes);</td>
</tr>
<tr>
<td>7 NextNode ← GetFirst(nNodes);</td>
</tr>
<tr>
<td>8 return NextNode;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation Time</td>
<td>1000 seconds</td>
</tr>
<tr>
<td>Number of events</td>
<td>100</td>
</tr>
<tr>
<td>Number of nodes</td>
<td>5-8</td>
</tr>
<tr>
<td>Data rate</td>
<td>80bps</td>
</tr>
<tr>
<td>Transmit power</td>
<td>160 dB re μ Pa</td>
</tr>
<tr>
<td>Minimum SINR</td>
<td>80dB</td>
</tr>
<tr>
<td>Neighbor holding time</td>
<td>100 seconds</td>
</tr>
<tr>
<td>Transmission range</td>
<td>50m</td>
</tr>
<tr>
<td>Volume</td>
<td>100m³</td>
</tr>
</tbody>
</table>

Table I: Parameter settings.
VI. CONCLUSIONS

We presented a location-free link state routing protocol for UASN. Our approach considers the characteristics of the communication channel to avoid communication void problems. Every node ranks the quality of the path that it offers toward the sink and generates beacon messages that include hop count, path quality toward a sink and pressure. Every node selects as next hop, a one-hop neighbor with the lowest hop count value as next hop for packet forwarding. In case of a tie, the greatest path quality node is selected. If the tie persists, then the lowest pressure node is selected. Our strategy is loop-free and comprises a recovery mode that kicks in when network topology changes. We implemented our routing protocol in ns-3 to conduct experiments.

VII. ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial support received from the following organizations: PWGSC contract # W7707-145688, Natural Sciences and Engineering Research Council of Canada (NSERC), Spanish Ministry of Science (project TIN2011-27076-C03-02 CO-PRIV ACY) and Ministry of Education of Mexico (SEP–PRODEP).

REFERENCES