Poster Abstract: Modeling the Underwater Acoustic Channel in ns2

Albert F. Harris III
Department of Computer Science
University of Illinois at Urbana–Champaign
aharris@cs.uiuc.edu

Michele Zorzi
Department of Information Engineering
University of Padova, Italy
zorzi@dei.unipd.it

1. INTRODUCTION

Underwater acoustic networks have the potential to support a wide variety of applications from facilitating communication between autonomous underwater vehicles to supporting the remote monitoring of underwater mining equipment or environmental conditions [1]. Even though underwater acoustics has been studied for many years, and communications technologies exist for underwater scenarios, interest on networking and protocol design in this environment is just beginning. As such, there is currently no standard underwater simulation module for any of the major network simulators. To accurately model underwater communication, channel, propagation, and interface models are required.

Since the underwater environment is different from its terrestrial counterpart [1, 7], it is likely that existing wireless modules cannot be easily reused, and specific underwater extensions will be needed. The underwater environment differs from the terrestrial radio environment both in terms of its modem energy costs, and in terms of the channel propagation phenomena. The underwater channel is characterized by long propagation times and frequency-dependent attenuation that is highly affected by the distance between nodes [7] as well as by the link orientation.

The main contribution of this work is an underwater propagation, channel, and interface model for the widely-used ns2 network simulator [5]. In this abstract, we give a brief description of its functionality and refer the reader to [3]. A challenge in building such a model lies in the fact that many of the features of the channel, such as bandwidth, depend on the distance between the two nodes and the orientation of the link. However, in ns2, the bandwidth is usually assumed to be fixed and used by layers higher than the propagation and channel model. Therefore, in underwater scenarios, bandwidth information has to be calculated and returned to the upper layers of the simulator, which is not the case for other radio models. The ns2 simulator divides the layers below the MAC layer into four components: Propagation, Channel, Physical, and Modulation. We provide modules for each layer, allowing protocol developers to concentrate efforts on the higher layers of the network protocol stack.

Presented as a poster in WUWNet '07

2. NS2 UNDERWATER MODEL

The ns2 simulator divides the channel and physical layer functions and characteristics into four components: Propagation, Channel, Physical, and Modulation. Figure 1 depicts this division, highlighting the characteristics within each component. In addition to distance-dependent attenuation, in underwater channels the signal fading is also affected by the orientation of the link. This feature is also modeled in the propagation component. The characteristics exported to other components of the ns2 model include the calculation of the received signal strength and the interference range of a signal. The primary function of the channel model is to handle propagation delay calculations and to make use of the functions from the propagation model. The physical layer tracks energy consumption metrics and also calculates the transmission times. Unlike in radio models, where the bandwidth is assumed to be constant regardless of the transmitter-receiver distance and therefore no information for other layers is required, in an underwater network the link bandwidth does depend on the link length, and therefore bandwidth information from the propagation layer of ns2 must be exposed to other components. Finally, the physical model calls the modulation model to calculate bit error probabilities given a received signal strength, and level of noise.

2.1 Underwater Propagation Model

In ns2, the Propagation model is responsible for calculating the signal strength at the receiver after attenuation is taken into account. It also calculates the interference range of a signal.

To use the underwater propagation model, it is only necessary to choose it in the TCL simulation script using the name "Propagation/Underwater".

The propagation model is used by the channel model to make collision and transmission error decisions; therefore, it does not need to calculate propagation delay or bandwidth. These functions exist in the channel model, which is described in detail in the following subsection.

2.2 Underwater Channel Model

The channel model in ns2 maintains the node lists used to calculate neighbor sets, collisions, etc. It is additionally responsible for calculating propagation delays. Essentially, the physical layer calls a *sendUp* function with a packet and a pointer to itself, and the channel model calculates neighbors that may be affected by the transmission as well as propagation delays and returns this information. Details on the exact functionality of the ns2 simulator can be found on the official website [5]. Aside from calling the appropriate propagation model functions, such as *getDist*, the ns2 channel model has to implement the propagation delay model as well, which is somewhat complex due to the dependency of the speed

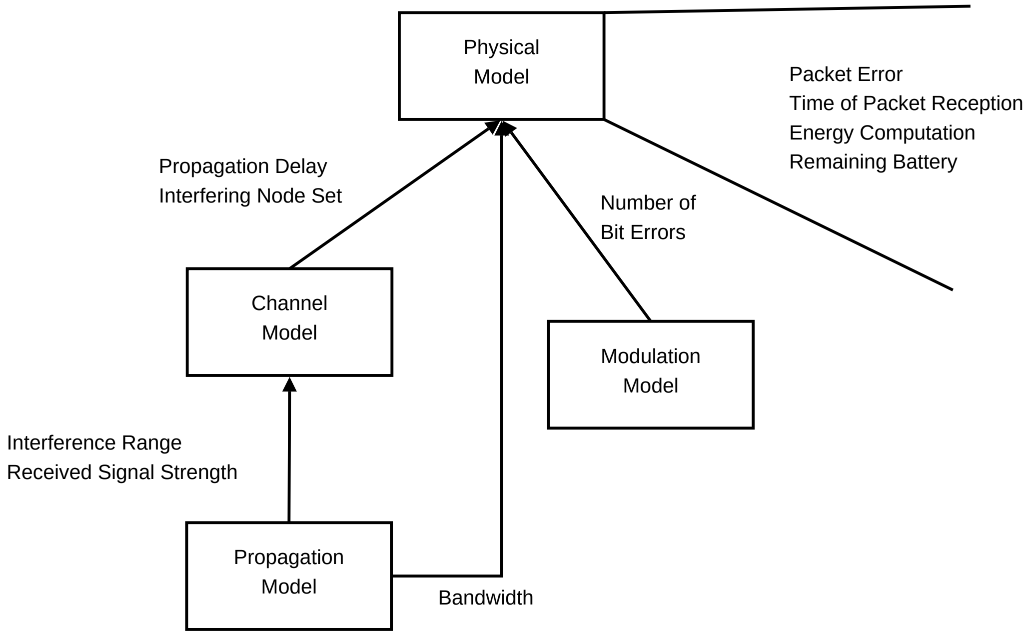

Figure 1: The ns2 channel and physical layer model

of sound on the depth of the water. In addition to the depth in the water, the propagation speed also depends on the temperature of the water, which in turn depends on the depth through a non-linear relationship. There are roughly five zones in which the temperature change in the oceans can be linearly approximated (see [9] for details).

To use the underwater channel model, it is only necessary to use the name "Channel/UnderwaterChannel" in the TCL script. There is only one bound variable to set in the channel model, the salinity value for the water used in the propagation delay calculation. This value defaults to the average salinity in the world's oceans (35 parts-per-million) [6] and can be set to some other value (*e.g.,* 32 ppm) as follows:

Channel/UnderwaterChannel set salinity_ 32

The physical layer model uses information from both the channel model and the propagation model to calculate transmission times, total delays, and the success or failure of packet reception. The physical model is described in detail in the following subsection.

2.3 Underwater Physical Layer Model

In ns2, the physical layer model calculates the final statistics used in the simulation with respect to packet reception, including packet error, transmission time, and propagation delay. For most of these calculations, calls are made to functions in the channel and propagation models. Additionally, information about energy costs associated with the physical interface are stored and used to calculate residual battery charge and transmission energy costs. We leave all the specific parameters of interface energy consumption as bound variables to be set by the user, since they depend on the specific hardware being modeled. Additionally, the receive signal strength threshold and the maximum transmit power levels are interface specific and are set through bound variables. The default sets of parameters for the maximum transmit power, receive threshold, and the interface energy consumption parameters are set to model the WHOI micromodem [2]. Also included is the parameters set to model the Teledyne-Benthos modem [8].

To use the underwater physical model, it is only necessary to use the name "Phy/UnderwaterPhy" in the simulation script. To set the maximum transmit power and the receive threshold, set the variables *Pt_* and *Pr_* respectively, both in dB re μPa.

2.4 Underwater Modulation Model

The modulation layer model relates the signal-to-noise ratio and interference to packet errors according to the modulation scheme used.

3. CONCLUSIONS AND FUTURE DIRECTIONS

The drive to design and test protocols for the underwater acoustic environment is growing steadily due to the desire to monitor and explore the world's oceans. However, no standard model is available for researchers to use in the simulation phase of their work. In this abstract, we have presented the design and implementation of our underwater acoustic model for the ns2 network simulator. Our model has four components: propagation, channel, physical, and modulation. We have described the critical functionalities of each of these layers (for a complete description see [3]). The ns2 model can be downloaded on the DEI SIGNET group website: http://www.dei.unipd.it/ricerca/signet/tools/ns_underwater.

We used the ns2 model in the design of a routing algorithm [4] that attempts to choose the optimal hop lengths as packets are routed through a multihop underwater network. The analysis of this protocol strongly depended on the bandwidth–distance relationship, as well as an understanding of the interference due to other nodes in the network and of the propagation delays. In addition to the use of the ns2 simulator, we used an implementation of the underwater channel previously developed in Matlab that, while able to accurately represent propagation and physical layer issues, failed to incorporate protocol issues such as collisions and multiple-access interference. The most significant point when comparing the results from the two simulators was that the energy savings due to the use of shorter hop distances was significantly underestimated by the Matlab simulator compared to the ns2 results, due to the absence of collisions and interference in the former. Using power control to reduce the transmission range also decreases the interference range. Without the entire model in the ns2 network simulator, our conclusions about the protocol performance would have been off by as much as 30%. This example result shows the usefulness of the ns2 model presented.

Work towards accurate modeling of the complex fading and multipath effects in the underwater acoustic environment would round out the simulation model. Current best practices use ray tracing; however, such techniques are often too computationally complex for inclusion in network simulators such as ns2. Therefore, approximations are needed to support PHY and MAC layer protocol development that takes into account these characteristics of the acoustic channel. More complex modulation schemes could be added to allow more realistic simulation of error tolerance in the channel as well as the true available bit-rates given the bandwidth. Additionally, models of additional acoustic modems could be included to facilitate energy consumption analysis for energy-aware protocol design.

Finally, a full protocol stack, including MAC, routing, and transport layers, is required to round out the ns2 underwater simulation suite. However, research in designing protocols for each of these layers is just beginning. We believe that the existence of simulation tools for underwater networks that properly capture the essential behavior of acoustic propagation in the water will provide a valuable instrument for protocol design and evaluations, and will greatly help promoting research in this area.

4. REFERENCES

[1] I. Akyildiz, D. Pompili, and T. Melodia. Underwater acoustic sensor networks: research challenges. *Elsevier's Ad Hoc Networks*, 3(3), 2005.

[2] L. Freitag, M. Grund, S. Singh, J. Partan, P. Koski, and K. Ball. The WHOI micro-modem: An acoustic communications and navigation system for multiple platforms. http://www.whoi.edu, 2005.

[3] A. F. Harris and M. Zorzi. Modeling the underwater acoustic channel in ns2. In *ACM International Workshop on Network Simulation Tools (NSTools)*, 2007.

[4] A. Harris III and M. Zorzi. Energy–efficient routing protocol design considerations for underwater networks. In *Proc. IEEE SECON*, June 2007.

[5] ns2 Network Simulator. http://www.isi.edu/nsnam/ns/.

[6] Office of Naval Research. Ocean Water: Salinity. http://www.onr.navy.mil/Focus/ocean/water/salinity1.htm.

[7] M. Stojanovic. On the relationship between capacity and distance in an underwater acoustic communication channel. In *Proc. ACM WUWNet*, pages 41–47, Los Angeles, CA, Sept. 2006.

[8] Teledyne. Teledyne-benthos modem. http://www.rdinstruments.com/nemo.

[9] The National Center for Atmospheric Research. Temperature of Ocean Water. http://www.windows.ucar.edu/tour/link=/earth/Water/temp.html&edu=high.