A Reservation MAC Protocol for Ad-Hoc Underwater Acoustic Sensor Networks

Leonard Tracy and Sumit Roy
University of Washington
Dept. of Electrical Eng., Box 352500 Seattle WA 98195-2500
{lentracy, sroy}@u.washington.edu

ABSTRACT

In this paper we propose a new MAC protocol for Ad-Hoc Underwater Acoustic Sensor networks that segregates the available bandwidth into a small control channel and a majority bandwidth main data channel. Reservations for main channel time are made by transmission of RTS packets on the control channel. The effects of channel segregation are explored and simulation results are presented. We find this reservation MAC provides an efficient means for underwater sensor nodes to relay data to gateway nodes in terms of both throughput and energy efficiency.

Categories and Subject Descriptors

C.2 [**Network Protocols**]: Protocol Architecure

General Terms

Design

Keywords

Underwater Acoustic Network, MAC Protocols, Reservation MAC, Sensor Networks

1. INTRODUCTION

Underwater acoustic networking is an increasingly important field of research. Long term monitoring of ocean properties via underwater observatories has become a necessity for many fields including oceanography and climatology. The Neptune project [1] is one such planned observatory that uses cabled networks for communications. Wireless communications is necessary to cost effectively increase the range and effectiveness of these observatories via networking to fixed sensors and autonomous underwater vehicles (AUVs). The pathloss endured by RF and optical communications underwater render them ineffective in communication at distances of more than a few meters, which makes acoustics the medium of choice. The acoustic channel, however, is characterized by long propagation delays, high delay spread, and frequency dependent fading. Application of terrestrial communication techniques to the challenging underwater environment often leads to less than adequate performance.

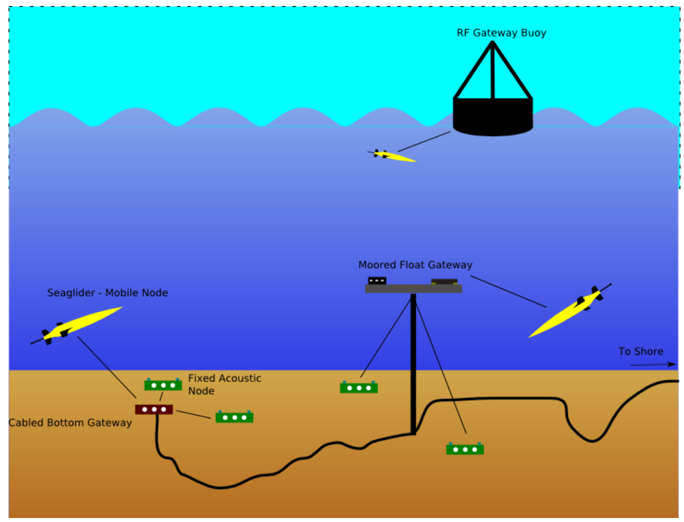

Figure 1: Common underwater network scenario

MAC protocols for underwater networks face a host of challenges not seen in their terrestrial counterparts. Long propagation delays amplify the penalty of handshaking protocols. Acoustic channels are severely band limited, usually on the order of a few kilohertz, resulting in data rates much lower than those expected in terrestrial networks. Additionally, while many terrestrial RF radios consume on the order of 1 Watt while transmitting, acoustic modems generally require transmission power an order of magnitude larger.

Many MAC protocols have been proposed to overcome these challenges. [2] attempts to reduce collisions via modified versions of the Aloha protocol. [3] also proposes an Aloha based protocol designed specifically for the WHOI Micromodem (a common acoustic modem for use in underwater networks) that introduces a backoff timer to mitigate collisions [4]. [5] proposes a hybrid protocol that combines TDMA scheduling and a contention based protocol to improve efficiency and throughput. These techniques have been met with mixed success, and are generally inefficient in networks with moderate to high traffic.

Many underwater protocols seek to use RTS/CTS handshaking in order to avoid collisions. [6] proposes an energy efficient protocol suited to networks with low traffic load by allowing nodes to sleep between transmission. [7] seeks to reduce the probability of collision by introducing slots to the FAMA protocol. [8] attempts to maximize throughput by adjusting the latency between the reception of a CTS packet and the transmission of its data packet. [9] also imposes a mandatory latency to allow potential interferers time to receive notification of an intended transmission, but also introduces MLP. MLP allows nodes to process and schedule future transmissions in an interlaced order as opposed to sequentially, negating much of the propagation delay penalty imposed by handshaking.

In this paper we assume an underwater sensor network that is facilitated by gateway nodes with out of band communications capability as shown in Fig. 1. These gateway nodes may be cabled bottom nodes which have wired network support to shore, or buoys that have RF capability and battery capacity significantly longer than neighboring non-gateway nodes. We envision that the rest of the acoustic network is made up by fixed and mobile acoustic only nodes. These devices, which require acoustic communications, are often significantly power limited, and the data generated at these devices is often destined for the provided gateway nodes. Sensor nodes may be arranged, however, such that there is no direct path to a gateway node, and routing via one or more acoustic only nodes is required. While some non-gateway to non-gateway communication is expected, this type of communication is assumed to be minimal and delay tolerant. We seek to optimize throughput and power efficiency of the acoustic network.

We propose a new MAC protocol termed Reservation Channel Acoustic Media Access Protocol (RCAMAC) which is based on RTS/CTS handshaking. Our MAC seeks to improve channel utilization and throughput over previous protocols via the introduction of a segregated channel for RTS packet transmission. By transmitting short RTS packets on an orthogonal low bandwidth control channel we can maximize utilization of the majority bandwidth main channel. We seek to attain performance benefits similar to those reported in [9] without requiring time synchronization or node placement information. We will first give a description of our proposed MAC protocol, then provide an analysis of the impact of segregating the available bandwidth, and finally present the results of our simulations. While we envision this protocol to be applicable in multi-hop, multi-gateway networks, in this initial work we will focus on single-hop, single-gateway networks.

2. MAC DESCRIPTION

The goal of this MAC is to make use of common underwater network topologies and a multi channel PHY to optimize scheduling. To this end, nodes will be classified into three types (1) Gateway, (2) Fixed acoustic only, and (3) Mobile. Gateway nodes are assumed to have high speed out of band communications (i.e. wired connectivity or RF wireless). Gateway nodes are also assumed to be non power limited. Non-fixed nodes (AUVs, UUVs, etc..) will be classified as mobile. We assume that data originates at non-gateway nodes and is destined for a gateway node. We seek to optimize the utilization of the available bandwidth and to minimize the power consumed in acoustic communication. High end to end latency is assumed to be tolerable. Nodes will be addressed such that other nodes can identify the sending node type from its address (for example: all fixed node

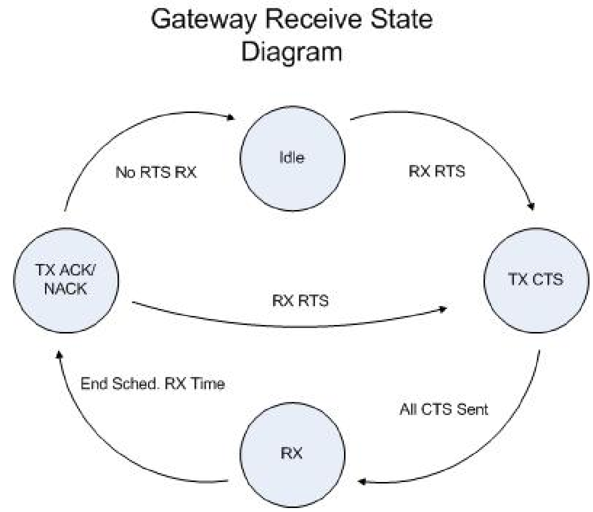

Figure 3: Gateway node RX process

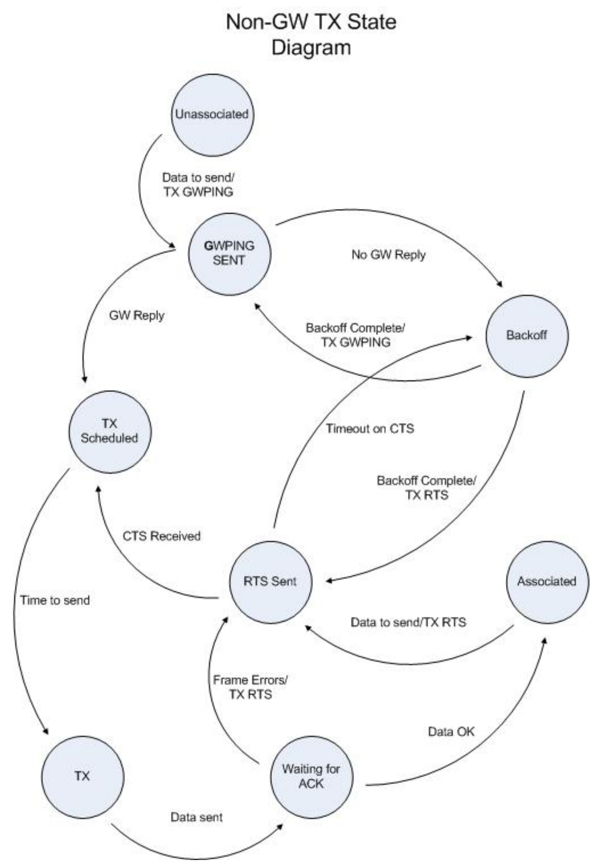

Figure 4: Non-Gateway node TX process

addresses begin with a 0 bit, gateway node addresses begin with a 10 and mobile node a 11). State diagrams of the GW receive process and non-gateway send process are shown in Figs. 3 and 4 respectively.

The channel will be divided into two orthogonal portions: A low bandwidth CTRL channel and a majority bandwidth MAIN channel. Non gateway nodes will reserve time on

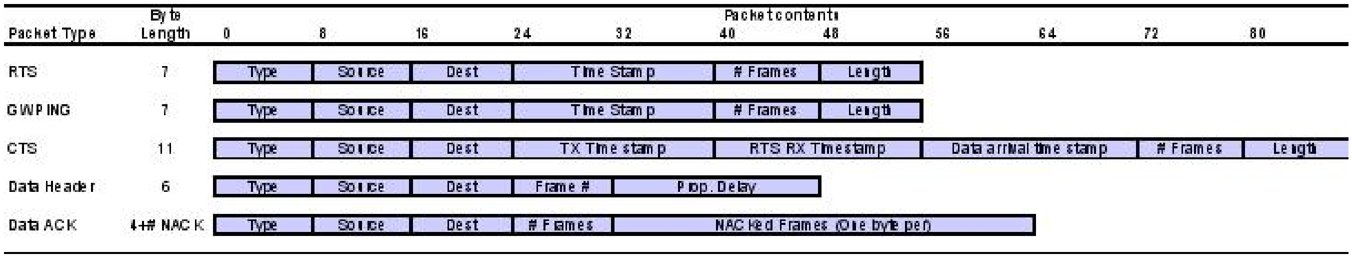

Figure 2: Contents of control packets

the main channel by transmitting RTS packets on the control channel. Gateway nodes will only transmit on the main channel. Channel segregation can be accomplished, for example, by reserving a small subset of subcarriers in an OFDM physical layer. The details of how the channel segregation is accomplished are not important to MAC layer operation. The assumption made by the MAC is that a node can TX on one or both of these channels at any time. A nearby transmission on one channel will not affect reception on the other channel. A node transmitting on the MAIN channel is assumed to be deaf to a transmission on the CTRL channel and vice versa (i.e. full duplex communication is not allowed as it is likely sensors will use a single transducer).

Non gateway nodes in range of a gateway will associate themselves with a gateway by transmitting a short GWPING packet on the control channel. The length and contents of control packets are shown in Fig. 2. The GWPING packet is an RTS packet with the type field adjusted to notify Gateways that this node is seeking association. Gateway nodes in range of the transmission will receive the packet and communicate amongst themselves via their out of band communications medium that they have received an association request. The gateway nearest the transmission (the first to receive the packet) will claim the new node by replying with a CTS packet on the main channel at its next available time. The association process is repeated at short intervals for mobile nodes to account for node movement in the network. If multiple gateways overhear a node's GWPING packet, the nearest gateway node must notify other in-range gateways of assigned time slots in order to avoid collisions.

While not transmitting, gateway nodes will monitor the control channel for incoming GWPING and RTS packets. If the gateway is idle when an RTS packet arrives, it will respond with a CTS on the main channel after a period of SIFS. We define SIFS to be a value which encompasses expected channel clearing time, required processing time to finish a reception and begin another reception or transmission, and a guard time to allow for timing errors introduced by quantization error and clock drift. If the gateway is currently receiving data packets, it will respond to each received RTS packet with a CTS packet after reception of currently scheduled data packets and their acknowledgements. Acknowledgements and CTS packets transmitted by gateways will take place on the main channel. RTS packets will include the number of data frames to be sent, the length of the data packets, and a timestamp. The corresponding CTS packets will also contain the number of frames and total transmission length, and additionally contain the time of RTS packet arrival, the timestamp for transmission of the CTS, and the desired time of arrival for the data. The transmitting node can then determine the clock offset between itself and the gateway and the transmission delay between the two nodes which allows it to transmit the data on schedule [10]. The MAC header attached to the data packet will also include the propagation delay between the two nodes to be used in optimizing scheduling of future transmissions.

In order to ensure a feasible schedule and interference free reception of CTS packets at transmitting nodes, the gateway must ensure a proper delay between the transmission of the CTS packet and the scheduled arrival time of data at the gateway. This can be assured, as mentioned in [9], by allowing for the maximum possible round trip time between the last CTS packet sent and the arrival of the first data packet. If, however, we make the assumption that the change in propagation delay between the gateway and its associated fixed nodes is negligible, then the gateway can log the propagation delay information for future reference and optimize the schedule. We accomplish this by ordering the CTS packets such that the packet with the farthest to travel (and those for which we do not have propagation delay information) are transmitted first and the CTS for the nearest node is transmitted last. Closer nodes can then be scheduled to begin transmission shortly after receiving the CTS packet. The gateway must only assure that incoming data packets are scheduled such that farther away nodes allow delay sufficient that their transmissions do not overlap at the gateway with transmissions from closer nodes.

Requesting node's RTS packets which are not received at the gateway due to collision or error must be retransmitted. As the length of each MAC cycle is dependent on the number of data frames included per RTS packet and the number of RTS packets correctly received in the previous cycle, the optimal timeout period varies for each cycle. We consider an RTS packet lost if no CTS is received in the next gateway transmission phase, or if there is no channel activity at all for some arbitrary timeout period RTSTO. RTSTO should be long enough that it is unlikely a node will attempt to transmit an RTS in the same cycle more than once. In order to mitigate the probability of future collisions we introduce an exponential backoff period after an RTS timeout. We set an integer paramater M, and after timeout randomly choose an integer n in the interval $n \in ([0, M])$. We then backoff for a period nT_{RTS} where T_{RTS} is the transmission time of an RTS packet. On initial transmission of the RTS we set $M = 1$, and then for each successive timeout we double M. On reception of a CTS packet, M is reset to 1. GWPING packets also follow the same backoff rule.

We also include an ARQ scheme for data transmission packets. Gateway nodes will respond at the end of a cycle with an acknowledgement for each transmitter in the

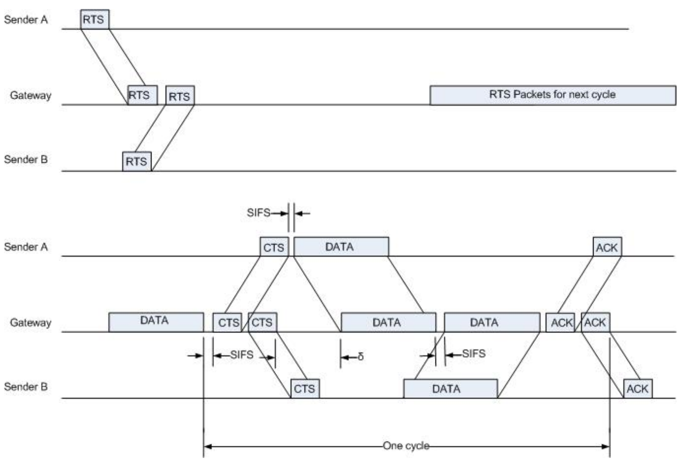

Figure 5: Typical cycle using dedicated control channel. *Top:* Control channel, *Bottom:* Main channel

preceding cycle. ACK packets will reply with the number of correctly received frames and explicitly request missing frames. In this way the acknowledgement scheme, within each reservation, resembles selective repeat.

We envision extending this protocol for more complicated multi-hop multi-gateway networks consisting of fixed nodes relaying data from outlying fixed and mobile nodes out of range of any gateway. However, for the remainder of this initial paper we will focus on analysis and simulation results of this protocol operating in a single hop network composed of fixed acoustic only nodes attempting to communicate with a single gateway. We will first look at long term utilization of this protocol, then provide analysis of the effects on channel segregation and throughput. Finally, we will provide results from extensive simulations.

3. LONG TERM AVERAGE CHANNEL UTILIZATION

The timing of our MAC is shown in Fig. 5. The main channel utilization over a single cycle of CTS transmission, data reception, and acknowledgement is

$$U_m = \frac{\gamma T_{data}}{T_{cts} + T_{data} + T_{ack} + 3SIFS + \frac{2\delta}{N}} \quad (1)$$

where T_x is the transmission delay of packet type x. γ is the fraction of useful data contained in a data packet (as opposed to headers), and N is the number of nodes scheduled in the transmission block. δ is the one way propagation delay allowed for CTS packet reception. We can also see from Fig. 5 that the receiver is available to receive RTS packets for a fraction of each slot equivalent to

$$W_{RTS} = \frac{T_{data} + \frac{2\delta}{N}}{T_{cts} + T_{data} + T_{ack} + 3SIFS + \frac{2\delta}{N}} \quad (2)$$

If we assume that packets arrive in the network as a poisson process with an arrival rate of λ corresponding to the offered load of the entire available bandwidth and that there are a very large number of nodes in the network, then packets will

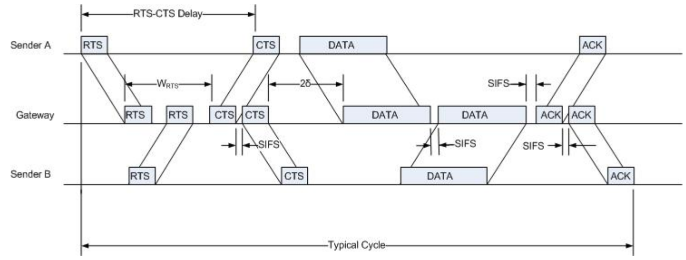

Figure 6: Common timing of hypothetical single channel protocol

arrive on the control channel with the rate

$$\lambda_c = \lambda \frac{L_{rts} R}{L_{data} R_c}$$

Where we have assumed that the rate of a node using the entire available bandwidth is R, the rates used for the control and main channels are R_c and R_m respectively, and that $R_c + R_m = R$. L_{rts} and L_{data} are the sizes of the RTS and data frames respectively. As nodes transmit RTS packets on the control channel using the Aloha protocol, the probability of correctly receiving a transmitted packet at the gateway is

$$P_s = W_{rts} e^{-2\lambda_c} \quad (3)$$

And we can see that the expected number of received RTS packets over the current cycle of transmission conditioned on the length of the current cycle is

$$E[N_{next}|N] = \frac{\lambda R (N T_{data} + 2\delta)}{R_m T_{data}} e^{-2\lambda \frac{L_{rts} R}{L_{data} R_c}} \quad (4)$$

If Eqn. 4 is greater than 2 for $N = 1$, then, assuming λ is constant, we can expect N to grow arbitrarily large in steady state. This condition amounts to putting a limit on the ratio R/R_c for a given set of packet sizes. When N is large, we can expect the normalized long term throughput to approach

$$S = \frac{R_m \gamma T_{data}}{R(T_{cts} + T_{data} + T_{ack} + 3SIFS)} \quad (5)$$

which gives an upper bound for performance of this protocol. The condition of Eqn. 4 is relaxed by the fact that we've allowed nodes to queue data and transmit multiple data frames per RTS/CTS exchange. Additionally, because nodes retransmit data, we cannot expect the offered load to remain constant. As such, we see in simulation that under these assumptions the required control channel bandwidth is much less than that predicted by Eqn. 4.

4. CHANNELIZATION EFFECTS

In order to determine the conditions under which this protocol exceeds a similar protocol operating over the entire available bandwidth, we consider the hypothetic timing in Fig. 6. We have allowed a round trip time between the transmission of the final CTS packet and the first arriving data packet as well as the final ACK packet to the next arriving RTS packet in order to ensure correct delivery of packets. We will assume that, like our proposed protocol, this is the delay between the nearest node transmitting in a cycle. The long term average normalized throughput of this

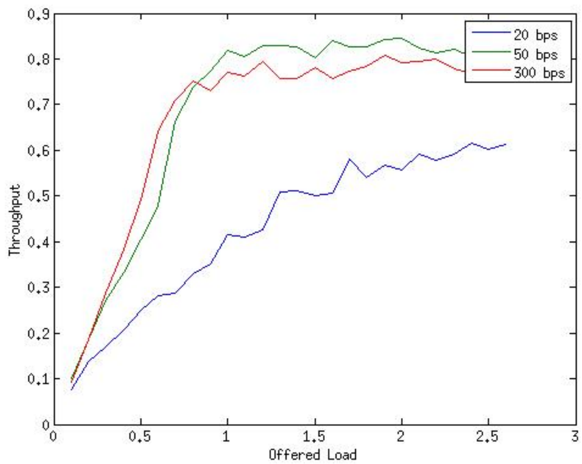

Figure 7: Throughput vs Offered load at several control channel rates

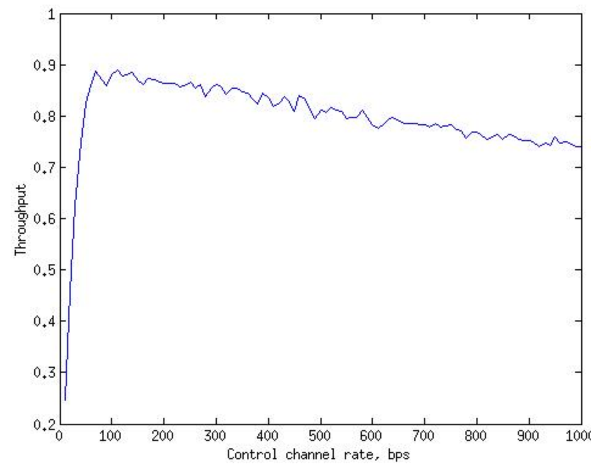

Figure 8: Throughput vs Control Channel Rate at offered load of 1.0

protocol will be

$$S_{1c} = \frac{\gamma L_{data}}{L_{rts} + L_{cts} + L_{data} + L_{ack} + R[(4 - \frac{2}{N})SIFS + \frac{4\delta}{N}]} \quad (6)$$

Where we have assumed that the delay period W_{RTS} in Fig. 6 is scheduled perfectly with N RTS packets.

Multiplying the upper bound throughput found in Eqn 5 and including the propagation delay term gives us

$$S_{2c} = \frac{\gamma R_m L_{data}}{R(L_{cts} + L_{data} + L_{ack} + R_m[(3 - \frac{1}{N})SIFS + \frac{2\delta}{N}]}$$

We can then say that the two channel protocol will perform better than the single channel protocol if $S_{2c} > S_{1c}$ or

$$\frac{R_m}{R} > \frac{L_{T2} + R_m[(3 - \frac{2}{N})SIFS + \frac{2\delta}{N}]}{L_{T1} + R[(4 - \frac{2}{N})SIFS + \frac{4\delta}{N}]} \quad (7)$$

Where we have made the substitutions

$$L_{T2} = L_{cts} + L_{data} + L_{ack}$$

and

$$L_{T1} = L_{cts} + L_{data} + L_{ack} + L_{rts}$$

If we substitute in the packet lengths outlined in Fig. 2 and then assume a total data rate of 4070bps available with 70bps dedicated to the control channel (which our simulations show in Section 5 to be the optimum value) we find that the theoretical best possible throughput delivered by our protocol is better for all values of $N < 388$ at $\delta = 0.5$. Given that underwater sensor networks are generally sparse, and that a single channel protocol must allow a W_{RTS} large enough for reception of the RTS packets, it is unlikely that $N \geq 388$ is attainable. It is also highly unlikely that the entire window of W_{RTS} will be perfectly filled with RTS packets. by If we require $Rc = 500bps$ and again assume perfect scheduling of RTS packets, then the two channel protocol theoretical best is still better for $N \leq 68$.

5. SIMULATION

We implemented our protocol in the popular freeware network simulator, NS2. We use the channel and propagation layers as described in [3]. The code for our simulation modules is available at [11]. For these simulations new PHY and MAC layers were developed that follow the assumptions we've made on two asymmetric orthogonal channels. As previously stated, we assume that from the total bandwidth R, the main channel will have portion R_m and the control channel will have portion R_c, and that $R_m + R_c = R$. A node can transmit on one or both of the channels at the same time or receive on one or both of the channels at the same time, but nodes cannot simultaneously transmit and receive. Unlike the previous work, we also make the assumption here, in order to focus on MAC behavior, that packets are only lost due to collision. We ran simulations of a 32 node single hop network. Nodes were uniformly placed in a 2km by 2km square region with a gateway node at the center of the region. Packets were generated according to a poisson distribution at nodes and were destined for the gateway node. In simulations we limit the gateway to scheduling 30 seconds of data packets, and set RTSTO to 35 seconds.

We first ran simulations with 1kB packets and altered the data rate of the ctrl channel from 100bps to 1000bps while keeping the main channel bandwidth fixed at 4kbps. For each control channel rate we simulated poisson traffic varying the offered load from 1.0 to 2.6. The results of these experiments are given in Fig. 7. Fig. 8 shows the throughput at an offered load of 1.0 at different control channel rates. We see from the plot that the optimum throughput is attained at a control channel rate of 70 bps at approximately 0.88.

Next we ran simulations holding the main channel data rate to 4kbps and the control channel at 70bps and varied the size of generated data frames from 100 bits to 1kB. The results appear in Fig. 9. We expect from Eqn. 4 that the achievable throughput will fall with smaller generated data frames, which is in agreement with the simulation results.

In order to measure protocol efficiency, we make the assumption that a node transmitting on the entire available bandwidth consumes P_t watts. A node transmitting on the

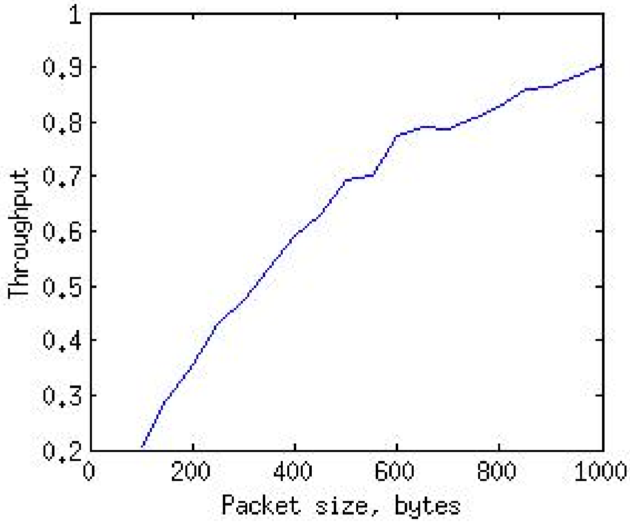

Figure 9: Throughput vs Generated Data Frame Size at offered load of 1.0

control channel consumes $P_c = R_c P_t / R$ watts and the main channel $P_m = R_m P_t / R$ watts. We also make the assumption that receive and idle powers are approximately equal, and that the majority of the power consumed is consumed in transmission. For the WHOI Micromodem operating in FH-FSK mode, this is a valid assumption [4]. We can then measure the power efficiency of the protocol at non gateway nodes, to be

$$eff = \frac{\sum_i \frac{R_m}{R} T_{data,i}}{\frac{R_c}{R}(\sum_j T_{rts,j} + \sum_k T_{gwping,k}) + \frac{R_m}{R}\sum_l T_{data,l}}$$

Where $T_{data,i}$ is the time spent receiving received packet i and $T_{rts,j}$, $T_{gwping,k}$ and $T_{data,l}$ are the time spent sending RTS, GWPING, and data packets respectively.

Over offered loads 0.1 to 2.5 at the optimal control channel rate of 50 bps, the network efficiency was 0.98. We have made the assumption that there is no error in computing the time to send reported in CTS packets which results in no data channel collisions. This is feasible by choosing a SIFS period large enough to account for any errors induced by clock drift.

6. CONCLUSION

Our presented MAC protocol provides an efficient method for moving data from sensor nodes to gateway nodes via ad hoc sensor networks. Our work presents great improvements over efficiency and throughput of other acoustic sensor network protocols and makes no assumptions on time synchronization or node placement information. While we have currently only analyzed and simulated single hop single gateway networks, we have begun to provide a framework for moving towards an efficient MAC protocol for more complicated multi hop ad hoc acoustic sensor networks.

7. REFERENCES

[1] (2000) Real-time, long-term ocean and earth studies at the scale of a tectonic plate: Neptune feasability study. [Online]. Available: http://www.neptune.washington.edu

[2] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "Aloha-based mac protocols with collision avoidance for underwater acoustic networks," *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pp. 2271–2275, May 2007.

[3] N. Parrish, L. Tracy, S. Roy, W. L. J. Fox, and P. Arabshahi, "System design considerations for undersea networks: Link and multiple access protocols," *Submitted to IEEE JSAC Special Issue on Underwater Wireless Communications and Networks*, 2008.

[4] 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," *Proc IEEE Oceans 2005*, pp. 1086–1092 Vol. 2, 2005.

[5] I. Kurtis B. Kredo and P. Mohapatra, "A hybrid medium access control protocol for underwater wireless networks," in *WuWNet '07: Proceedings of the second workshop on Underwater networks*. New York, NY, USA: ACM, 2007, pp. 33–40.

[6] V. Rodoplu and M. K. Park, "An energy-efficient mac protocol for underwater wireless acoustic networks," *OCEANS, 2005. Proceedings of MTS/IEEE*, pp. 1198–1203 Vol. 2, 2005.

[7] M. Molins and M. Stojanovic, "Slotted fama: a mac protocol for underwater acoustic networks," *OCEANS 2006 - Asia Pacific*, pp. 1–7, 16-19 May 2007.

[8] B. Peleato and M. Stojanovic, "A mac protocol for ad-hoc underwater acoustic sensor networks," in *WUWNet '06: Proceedings of the 1st ACM international workshop on Underwater networks*. New York, NY, USA: ACM, 2006, pp. 113–115.

[9] X. Guo, M. Frater, and M. Ryan, "An adaptive propagation-delay-tolerant mac protocol for underwater acoustic sensor networks," *OCEANS 2007 - Europe*, pp. 1–5, 18-21 June 2007.

[10] A. A. Syed and J. Heidemann, "Time synchronization for high latency acoustic networks," *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pp. 1–12, April 2006.

[11] (2007) Fundamentals of networking lab. [Online]. Available: http://ee.washington.edu/research/funlab/uan