Performance evaluation of underwater MAC protocols: From simulation to at-sea testing

Chiara Petrioli and Roberto Petroccia
Dipartimento di Informatica
Università di Roma "la Sapienza"
Roma, Italy
E-mail: {petrioli,petroccia}@di.uniroma1.it

John Potter
NATO Undersea Research Centre
La Spezia, Italy
E-mail: Potter@nurc.nato.int

***Abstract*—Many MAC protocols have been proposed for underwater sensor networks, usually variants of well-known terrestrial approaches. Although performance comparisons among different MAC protocols have been estimated by simulations, e.g. [1], [2], no extensive comparison has yet been performed by means of at-sea experiments. Simulations can only capture a subset of the total environmental variability, resulting in an approximate and generally simplified model of the acoustic channel and its dynamics. Moreover, they do not generally capture constraints introduced by the actual hardware and this can significantly impact the overall protocol performance (e.g., limitations in the packet format and size, latency introduced by the hardware, control overhead associated with a given acoustic modem operation, etc.). For these reasons, at-sea experiments are needed to validate not only the relative performance of different classes of MAC protocols but also the validity of the simulation process itself. We have developed a framework to seamlessly simulate, emulate and test (at-sea) a variety of communication protocols. Three candidate underwater MAC protocols (CSMA, T-Lohi and DACAP) have been implemented on our framework (representing simple, intermediate and fully negotiated protocols). We conducted various tests in the waters surrounding Pianosa island during September 2010, comprising both single and muti-hop messaging, under various types of application loads. We then simulated exactly the same scenarios and settings in order to compare simulation and at-sea trial results. We show that if an inadequate acoustic channel model is used, or (even more importantly) if the overheads and delays due to the specific hardware are not included, there is a significant gap between actual at-sea and simulation results. Once accurate models for the channel and the modem are introduced into the simulator, a significant reduction in the gap is achieved. Moreover, we show how overcoming some of the limitations of commercial acoustic modems is expected to result in much better system performance in terms of throughput efficiency and packet latency.**

***Index Terms*—Underwater acoustic networks, MAC, simulation, ns2-Miracle, WOSS, underwater wireless sensor networks, sea trial testing.**

I. Introduction

UnderWater Sensor Networks (UWSN) have become an important area of research with potential practical impact on a host of different applications, ranging from monitoring and discovery of the marine environment to remote control of submarine oil extraction, underwater safe CO_2 storage, etc. Low-cost, (quasi) real-time, medium/large scale monitoring systems are now possible through the deployment of underwater wireless sensor nodes equipped with acoustic modems. Given the challenges posed by the very specific environment (long propagation delay, low bandwidths, slow power signal attenuation, etc.), solutions for terrestrial wireless (sensor) networks cannot be directly deployed in UWSN, and novel protocol stacks are required for both single-hop and multi-hop communications. Particular importance is given to the design of new Medium Access Control (MAC) protocols that account for the high propagation delays, the low bandwidth available and the many challenges posed by the acoustic signal propagation.

Recently, many solutions for underwater MAC have been proposed. They are typically based on variants of successful terrestrial approaches. A coarse grain classification of existing channel access methods may be used to group them into Aloha based [3], carrier sensing solutions (e.g., CSMA [4], DACAP [5]), TDMA based schemes [6], [7], and CDMA based approaches [8]. Hybrid schemes have also been proposed [9]. Although a performance comparison among some sets of the best performing schemes have been made by means of simulations [1], [2] no extensive comparison has been performed so far by means of at-sea experiments. Moreover, existing simulators only approximately model the acoustic channel and the channel dynamics and do not capture constraints introduced by the hardware. This paper shows that these factors can significantly impact the overall protocol performance (e.g., via limitations on the packet format and size, delays introduced by the hardware, control overhead associated to a given acoustic modem operation, etc.) Therefore, at-sea experiments are needed to validate not only the relative performance of different classes of MAC protocols but also the validity of the simulation process itself.

The objectives of our work are twofolds; Firstly, we want to perform a thorough experimental evaluation of

978-1-61284-4577-0088-0/11/$26.00 ©2011 IEEE

some of the best known MAC protocols to assess the impact of different design choices and the pros and cons of exchanging higher amounts of control information to limit collisions. The goal is to reach an understanding of which protocol is likely to perform better in a given environment and application (considering several realistic scenarios and topologies). Various scenarios have been investigated to date, including static single-hop, mobile single-hop and static multi-hop. The number of nodes involved in the different experiments ranges from 4 to 6 incoherent FSK Micro-Modems [10]. Our second objective was to quantify, and reduce as much as possible, the gap between simulation results and at-sea experiments.

To reach these two objectives we compared extensive experimental data obtained running CSMA [11], T-Lohi [12] and DACAP [5] protocols during the ACommsNet10 experiment in Pianosa during September 2010, with simulation results obtained for the same set of protocols and for the same deployment scenarios obtained from an NS2-based simulation framework. This paper reports this research and is organized as follows; Section II begins by briefly describing the three MAC protocols we chose for evaluation, the framework on which these protocols were implemented, the experimental scenarios and performance metrics used in the sea trials and the modem hardware used at sea. The performance of the selected MAC protocols is presented in Section III. Our conclusions are presented in Section IV.

II. Performance evaluation

In this section we briefly review the MAC protocols we selected for our comparative performance evaluations. We also describe the approach we have followed for simulating and testing these protocols at sea. Finally, we discuss the experimental scenarios we used and the performance metrics we chose.

A. Selected MAC protocols

The three MAC protocols chosen for investigation were CSMA, T-Lohi and DACAP. All three protocols were tested in a variety of topologies and application loads in the waters surrounding Pianosa island, on the west coast of Italy, during September 2010.

(Readers familiar with these protocols may wish to skip this section.)

- CSMA (Carrier Sensing Multiple Access) is a well-known at sea protocol for channel access [11] that has the advantage of a low overhead, because it does not perform extensive handshaking to avoid collisions. When a node has a data packet to transmit, it first checks whether the channel is idle or busy. If idle, the node starts the packet transmission immediately. If the channel is busy, the node delays the transmission according to an exponential backoff mechanism. We consider two versions of this protocol. The first is the following: A node that transmits a data packet receives no feedback about whether the intended recipient has received it or not. The second adds robustness by having the destination node acknowledge (ACK) the data reception. If the ACK is not received by the sender within a given time (set to $2 * \text{Delay} + \text{ackTime}$), the data packet is re-transmitted either until successful reception (each time choosing the backoff time in an interval twice as long as the previous one) or until the maximum number of retries has been reached. Here, Delay is the transmission time between source and destination. Its value is initially set to maxDelay and successively reset by the nodes to a value computed according to the (estimated) distance between source and destination (which is based on the time difference between data packet transmission and ACK reception). The backoff time is chosen randomly and uniformly in $[0, T]$, where $T = 2^{\text{txRetry}}(2\text{maxDelay} + \text{dataTime} + \text{ackTime})$. Where dataTime and ackTime are the times needed to transmit a DATA packet and an ACK packet respectively. If no ACK is used ackTime is assumed to be zero. Retransmission of the same packet stops after a predefined number of tries. Moreover, when a node overhears a DATA packet on the channel and ACKs are used it goes into backoff mode, thus allowing the transmitter to correctly receive the ACK packet and the receiver to forward (if required) the data it has just received.
- T-Lohi. Tone Lohi is a protocol for single-hop underwater networks defined in [12] that uses a type of weak negotiation to reserve the channel. When a node has a data packet to transmit, it starts a reservation period (RP). An RP is made up of a certain number of slots called contention rounds (CRs). During a CR, the would-be sender transmits a short control packet (tone packet) to inform other nodes about its desire to access the channel. It then listens to the channel to detect if other nodes also have data packets to send. Each node contending for the channel counts how many other nodes do the same, based on the number of tone packets received during the CR. If no other tone is heard during the CR, the node seizes the channel. Its RP is over and it transmits the data packet. If contention occurs, the contenders back off for a number of CR chosen randomly and uniformly between $[0, N]$ where N is the number of competitors. A node RP continues until successful channel access. Notice that nodes do not need to be synchronized. Each node that has data packets to send starts its own RP

for channel access and transmission, independently of other nodes. The duration of a CR is set so that a node has enough time to detect as many contenders as possible. Of the many flavors of T-Lohi described in [12] we consider the most aggressive, i.e., the one that maximizes the channel utilization. In the aggressive T-Lohi the CR lasts for the time needed to transmit a tone packet plus the maximum anticipated propagation delay.

- DACAP (Distance Aware Collision Avoidance Protocol [5]) uses the formal RTS/CTS handshake to reserve the channel for packet transmission. More specifically, when a node has a data packet to send, it checks the channel, and if the channel is idle it transmits an RTS. Upon correctly receiving an RTS, the destination node replies immediately with a CTS and then waits for the data packet. DACAP adapts to the underwater channel characteristics by using a warning mechanism, as follows. If while waiting for a data packet, a destination node overhears a control packet for some other node, it sends a very short WARNING packet to its sender. Upon receiving a CTS, a sender waits for some time, T_{warning}, before transmitting the data packet. If it overhears another control packet or receives a WARNING packet from the destination during this time, the sender node aborts transmission. The length of the WARNING time depends on the distance between the source and destination, which the sender can learn by measuring the RTS/CTS round-trip time. When the receiver overhears an RTS and sends a warning, it does not know whether the warning will reach the sender in time to have it abort the transmission. Since a data packet can still arrive, the receiver must continue listening to the channel even after having sent a warning. For this reason, the WARNING time is defined as the minimum waiting period between receiving the CTS and sending the data that guarantees absence of harmful collisions. Potential interferers are blocked as usual in RTS/CTS schemes.

B. *Performance evaluation framework: from simulations to sea trials*

We have developed a framework to seamlessly simulate, emulate and test communication protocols in water [13]. The framework is based on an extended and improved version of ns2-Miracle [14]. To accurately model the underwater channel when performing simulations, the framework is connected to the Bellhop propagation simulator [15] via the WOSS [16] interface. When working in simulation mode, one PC runs a single ns2-Miracle process to simulate an entire network. When emulating the network, several ns2-Miracle processes have to be run. Each of them emulates the behavior of a single node, based on the selected protocol stack [13]. To allow the interaction between ns2-Miracle and real hardware the ns2-Miracle protocol stack is connected to the modem through a specific driver that implements the code used by ns2-Miracle to handle all the specific functionalities of the particular modem. Different modems can have different functionalities, requiring the development of different drivers. In order to perform at-sea testing, also with nodes which could not be connected to a PC (e.g., AUVs or static underwater nodes which were not cabled), we successfully ported ns2-Miracle to Gumstix [17], a physically small, energy efficient, inexpensive but high performance embedded device which can be easily included in the underwater node casing. We have also successfully interfaced Gumstix to several commercial acoustic modems, including the FSK Micro-Modems used for the current experiments.

This framework allows us to use the same code to simulate, emulate or test in water a given protocol (comparing results obtained in the three cases using exactly the same code and parameter settings).

C. *Experimental scenarios and performance metrics*

For our comparison we first performed tests at sea, where each node was running its own instance of our framework. Then we evaluated exactly the same scenarios, this time running the framework in simulation mode (simulating the entire network on a single PC): we re-used exactly the same programming code and the same parameter settings as during the sea trial.

The sea trials were performed in the waters surrounding Pianosa island, part of a protected marine park on the west coast of Italy, during September 2010. A full environmental impact assessment was carried out and approved by the governing body (the Parco Nazionale Archipelago Toscano). The coordinates in Table I refer to the node locations during the sea trial (Figure 1). Three static nodes (designated M1, M2 and M3) were cabled to shore for power supply and information transmission. During the tests the three nodes were connected to a PC running our framework (it was running three instances of our framework, one for each node).

One WHOI gateway buoy, equipped with a Gumstix, was moored at GW1 or GW2 depending on the specific test performed (single-hop or multi-hop). Different kinds of mobile nodes were used: 1) Two eFolaga AUVs (produced by Graaltech [18]) equipped with Gumstix; 2) The NURC CRV Leonardo, with a PC in the laboratory on board connected to the acoustic modem (deployed through a moonpool) and controlled via a radio link from land; 3) a “MANTA” portable modem system devised by the University of Porto, consisting of an acoustic modem, a small PC and a radio interface. This was carried around

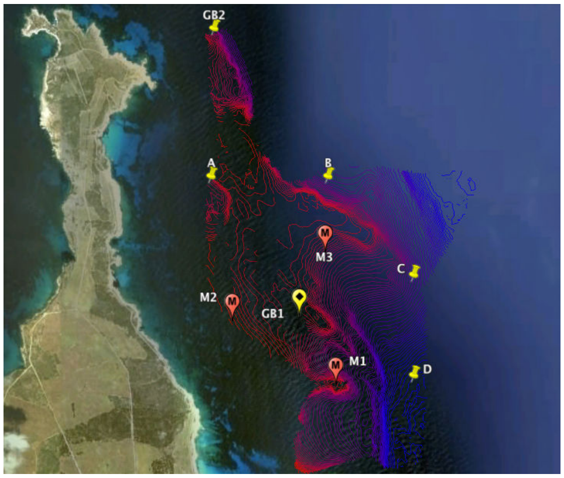


Figure 1. Node locations

Table I
LOCATIONS USED DURING THE SEA TRIAL

ID	Latitude	Longitude	Depth
M1	$42°35'44.52''N$	$10°6'19.08''E$	21m
M2	$42°35'58.92''N$	$10°5'47.40''E$	15m
M3	$42°36'14.40''N$	$10°6'15.84''E$	30m
GW1	$42°36'0.00''N$	$10°6'7.92''E$	5m
GW2	$42°37'3.41''N$	$10°5'41.06''E$	5m
A	$42°36'30.00'N$	$10°5'40.19''E$	-
B	$42°36'30.00''N$	$10°6'15.98''E$	-
C	$42°36'7.74''N$	$10°6'42.25''E$	-
D	$42°35'45.03''N$	$10°6'42.25''E$	-

by a Rigid Hull Inflatable Boat (RIB) and controlled in the same way as the modem aboard CRV Leonardo. All the devices were equipped with the incoherent FSK WHOI Micro-Modem, without co-processor.

The following scenarios were considered to evaluate protocol performance:

- **Static single-hop scenario**: Four nodes were located in M1, M2, M3 and GW1. Nodes in M1, M2, and M3 generated data packets that were transmitted to the sink, located in GW1.
- **Static multi-hop scenario**: Nodes were located in M1, M2, M3 and GW2. Nodes in M1 and M3 generated data packets, which were transmitted to the node in M2. M2 then relayed each packet to the sink at GW2.
- **Mobile single-hop scenario 1**: Five nodes were located at M1, M2, M3, GW1 and on the CRV Leonardo. CRV Leonardo sailed along the line $A-B-C-D$. In this scenario, nodes at M1, M2, M3 and the node on the CRV Leonardo generated data packets. The node at GW1 served as a sink.
- **Mobile single-hop scenario 2**: Six nodes were used. Three fixed nodes were located at M1, M2, M3. Three mobile nodes were also used (2 eFolagas and one modem on a RIB). One eFolaga moved from M1 to the middle point between M2 and M3, another eFolaga moved from M1 to M3 and the node on the RIB moved around points M1, M2 and M3. The node in M2 was working as a sink while the other nodes generated data packets.

In all the above testing scenarios, traffic was generated according to a Poisson process with aggregate (network-wide) rate λ packets per second. We define the normalized packet rate as $\overline{\lambda} = \lambda T_{\text{data}}$, where T_{data} is the transmission delay related to the data packet. The normalized packet rate was varied in the range 0 to 0.22 packets per packet time.

We have imposed a limit to the number of packet re-transmissions, experimentally tuning this value for each protocol. Our implementation of CSMA mandates discarding a packet after 4 failed transmission attempts, or 4 failed attempts to access the channel.

A similar constraint holds for DACAP concerning RTS packets: After 7 attempts to access the channel, or after 7 failed retransmissions, a data packet is discarded. For T-Lohi, a data packet is discarded after 7 attempts to access the channel, or after 7 consecutive transmissions with multiple contenders.

We have also limited the number of packets that can be stored in the nodes queue to 50. When the queue is full and a new packet arrives the oldest packet in queue is discarded. In this way nodes do not fill their buffers with old information.

D. Performance metrics

The following metrics have been used to assess protocol performance.

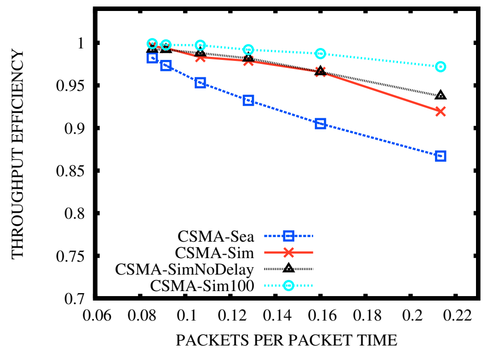
1) Throughput efficiency (Th), defined as the ratio between the bit rate delivered to the sink (correct bits) and the bit rate offered to the network, $N_b\lambda$.

2) End-to-end latency (L), defined as the average time between data packet generation and data packet reception at the sink.

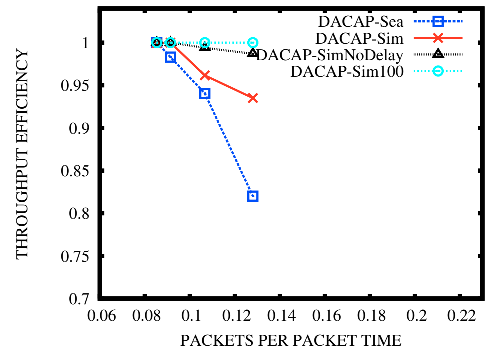
3) Data attempts (D), defined as the number of times nodes access the channel before a packet is successfully delivered to the next hop relay (both channel busy and packet retransmission are considered). For CSMA only data packets are considered while for DACAP and T-Lohi also RTS (R) and Tone (T) packets are considered. When displaying both data and control packets channel accesses in the same figure we named this metric "number of channel accesses".

E. FSK WHOI Micro-Modem features and their impact on protocol implementation

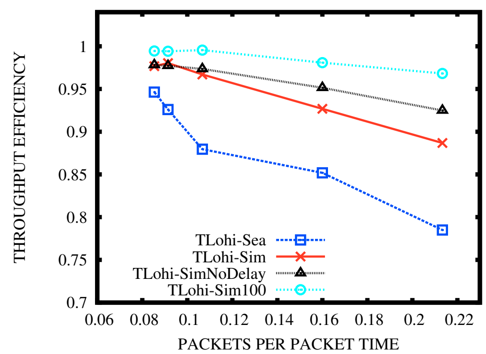
According to WHOI Micro-Modem specifications [10], FSK Micro-Modem transmission power is



(a) CSMA



(b) DACAP



(c) T-Lohi

Figure 2. Throughput efficiency results for the static single-hop scenario

Table II
METRICS ABBREVIATIONS USED IN SECTION III

Th	Throughput efficiency
L	End-to-end latency
D	Data attempts
R	RTS attempts
T	Tone attempts

180dB re 1 microPascal at 1m and the carrier frequency is centered at 24kHz. The transmission bit rate is 80bps ($T_{\text{data}} = 3.2$s) In order to communicate to the modem, a serial line connection has to be used with a baud rate of 19200. The FSK WHOI Micro-Modem does not provide the possibility of choosing the data packet size. Each packet has a size of 32Bytes and, although the transmission delay (T_{data}) is of 3.2s the actual time to transmit a packet is much longer: 0.87s are needed for the preamble (training sequence, header modulation etc.), 0.7s is the delay for the packet header and about 1s is needed to exchange control information before actual data transmission. Therefore, using the FSK WHOI Micro-Modem introduces a delay of about 2.6s for each data packet, corresponding to an increase in transmission delay, over what expected based on the data packet payload and header nominal sizes, of about 80%. The Micro-Modem does provide the option to use "mini packets", which have the capability to store 13bits of information and experience a total transmission delay of around 1s (including all the overhead). In order to reduce the overhead associated to the different MAC operations we have used mini packets to implement RTS, CTS, Tone and ACK packets. Even using the mini packets to reduce the overhead, control packets embedded in mini packets are much longer than what would be possible with dedicated control packets. This in turn affects the performance of the MAC protocols, impairing especially those MAC protocols which rely on significant amounts of control information for their operations. This explains why, among the selected MAC protocols, we shall see that DACAP experiences the most significant performance degradation with respect to theoretical performance.

In order to reduce as much as possible the gap between experimental and simulation results, we have introduced in our simulator an accurate description and estimation of all the delays, overhead and settings of the actual modem. In this way the collected experimental data can more accurately be compared with the simulation data for the same scenarios. Our simulator uses the Bellhop [15] channel model which has been fed with representative environmental data (bathymetry, sound speed profile, sea-floor sediment) for Pianosa island during September, when the experiments were conducted.

III. Results

Experiments involving only static cabled nodes and the WHOI gateway buoy were much easier to manage than when using mobile nodes. For the static experiments, the main constraint was the need to re-charge the gateway buoy once every couple of days. Therefore, we were able to run several static node experiments at different times of the day, collecting results for all the considered protocols and traffic loads. Unfortunately, after about ten days of operation the radio link of the gateway buoy failed. This cut short our experimental programme and we were not able to collect the results for DACAP with higher traffic load or to run all the envisioned tests for the multi-hop case. Tests involving mobile nodes were much more challenging to manage, mainly due to the higher number of people involved in operating eFolagas, RIBs and the CRV Leonardo. The available testing time was therefore more limited than for the static tests and the collected results are consequently less numerous.

For each protocol we have four different sets of results:

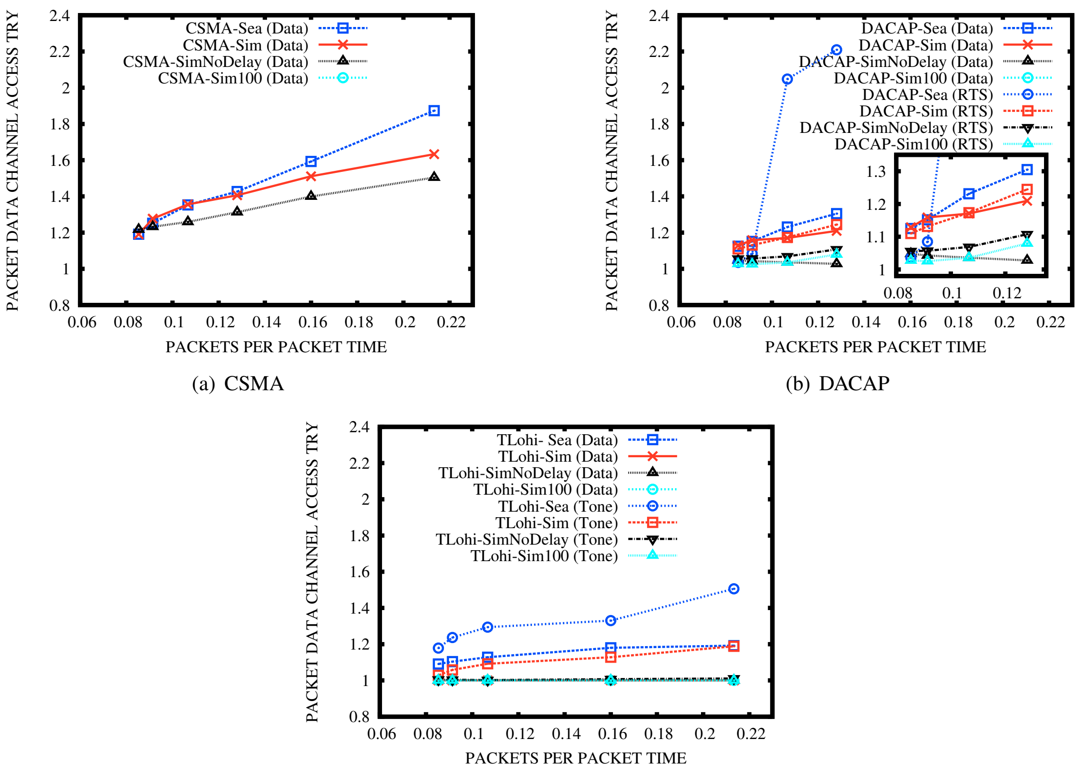
- **Sea:** Data collected during the sea trial.
- **Sim:** Data obtained simulating the sea trial scenarios, fully accounting for FSK WHOI Micro-Modem features and operation.
- **SimNoDelay:** Data obtained simulating the sea trial scenarios, considering the same 32Bytes packet size supported by the FSK WHOI Micro-Modem, but ignoring the delays related to the modem and Gumstix operations.
- **Sim100:** Data obtained simulating the sea trial scenarios, accounting for delays and overhead of the real acoustic device, but considering a data packet size respectively of either 100Bytes or instead the 32Bytes supported by the FSK Micro-Modem.

Results for the static single hop scenario are presented in Figures 2 to 4. Figure 2 shows the throughput efficiency achieved by the three protocols that we tested. We can see that in all cases, the at-sea experiments achieve a lower throughput efficiency than estimated by simulations. There are several reasons for this. Firstly, the simulator is not able to capture short-timescale changes in the environment (such as fading) that sometimes impair communications. A great deal of important oceanographic processes are thus ignored, including changing wind and weather conditions (that affect the heat content and stratification of the water column), and the passing of internal waves (that cause random focusing of acoustic eigenrays).

Moreover, the FSK WHOI Micro-Modem cannot transmit continuously but has to be limited to a duty cycle of about 50%. Otherwise the modem may require reinitialization, consequently failing to receive packets for short periods of time. Although the data packet generation rate was lower than the theoretical limit that should have been safely supported by the modems, we experienced several reinitializations, especially when running DACAP which makes larger use of control packets and at higher loads. This contributes to the DACAP results showing a larger gap between simulated and experimental performance. When using CSMA and T-Lohi, which are less affected by multiple control packet transmissions, the difference between at-sea data and simulated performance is moderate. CSMA experiences a simulated throughout efficiency that is within 7% of the throughput efficiency measured at sea. The gap for T-Lohi is about 11%. For DACAP, we noticed that simulated and experimental performance are close when the traffic load is low. As soon as the traffic load increases, thus challenging more modems, the gap in performance between simulations and experiments increases. The simulated DACAP performance delivered about 13% more packets than experiences at sea at $\overline{\lambda} = 0.128$.

We can better understand the differences between at-sea data and simulated performance by looking at the number of channel accesses needed to correctly transmit a data packet and an RTS or Tone packet (Figure 3). Figure 3 shows that simulation results are able to reproduce quite well what happens in reality for CSMA and T-Lohi, explaining why such protocols also show very small discrepancies between simulated and experimental packet latency (Figure 4). For DACAP, the simulated and experimental performance are quite similar in terms of number of data packets retransmissions. What differs significantly is the number of channel accesses a node needs to perform before it is able to complete the initial handshake needed to acquire the channel (channel accesses needed during the RTS phase). The number of RTS attempts measured experimentally is almost twice the value predicted by the simulations. About 45% of such channel attempts are due to the fact that the node finds the channel busy and 55% to the fact that the transmitting node fails to receive a CTS from the intended destination.

Moreover, we see that in simulation the percentage of ACK packets that are not correctly received by the destination is close to zero. When real acoustic devices are used, however, a significant percentage of ACK packets get lost. The percentage of lost ACKs is about 10% for CSMA and 7% for DACAP. It is important to notice, however, that loosing ACKs more significantly impairs the performance of a MAC protocol like DACAP



(a) CSMA

(b) DACAP

(c) T-Lohi

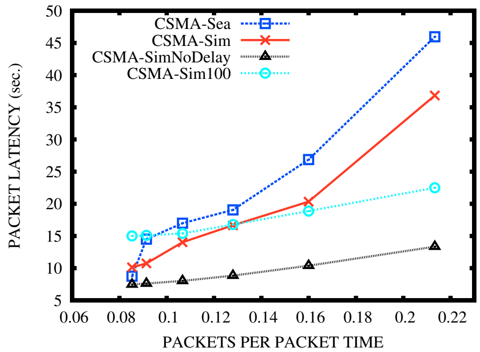
Figure 3. Data attempts results for the static single-hop scenario

than CSMA, because for each missed ACK the entire sequence of RTS/CTS/DATA/ACK packets has to be re-transmitted. Having more data packets in the network at the same time increases the probability of further collisions and re-transmissions, in turn increasing the number of channel accesses needed to advance a packet in the networks. The end result is an increase in end-to-end latency. Figure 4 shows the average end-to-end latency per packet. T-Lohi, without using ACK packet and not being affected by missed ACK packets and re-transmissions, provides a shorter latency. The downside is that there is no guarantee of packet delivery. CSMA and DACAP, instead, are more affected by packet re-transmission: CSMA due to the exponential backoff which includes long data packet transmission delays; DACAP due to the delays introduced by control packets.

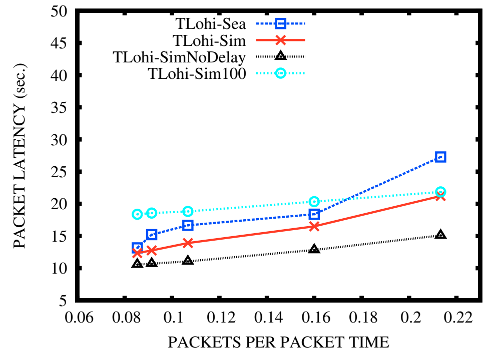
The fact that data packets may fail to be successfully delivered, even when the channel has been acquired, results in different variants of DACAP delivering different levels of performance when used at sea. In [1] simulation results suggest that we may wish to avoid the use of ACKs for the DACAP protocol in a single hop scenario. In this case, the use of RTS/CTS packets is enough to correctly reserve the channel and the use of ACK packets introduces a higher overhead and delay without any apparent benefit. In situations where we have an uncertainty concerning data packet delivery (introduced by vagaries of the channel) and, most significantly, by temporary modem reinitializations, this is no longer true. We have performed at-sea DACAP tests without using ACK packets to investigate this possibility. The results, in terms of packet latency, are 35% better than with the ACK version but the toll to pay is a 10% lower throughput efficiency.

The discussion above already provides evidence of the impact of actual acoustic modem features on overall performance. The two obvious questions now are whether significant performance improvements could be achieved by making minor changes (tuning) the modem parameters, and how critical it is to model all aspects of modem operations to design an accurate simulator.

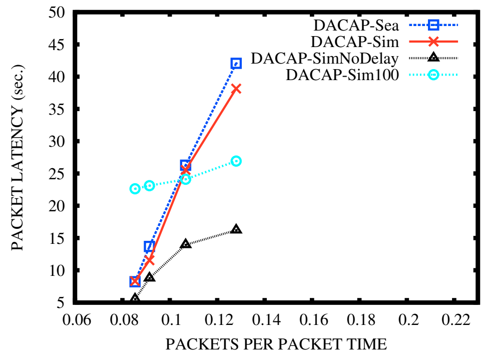
To investigate whether different modem parameter settings could give a significant performance improvement we performed simulations keeping everything fixed apart from the packet size, which was set to 100Bytes rather than the 32Bytes currently supported by Micro-Modems. Comparing the throughput efficiency for Sim and Sim100 we can see the benefit of adopting larger data packet sizes, which result in lower overhead. Note that increasing the transmission delay can increase



(a) CSMA



(b) T-Lohi



(c) DACAP

Figure 4. End-to-end latency results for the static single-hop scenario

packet latency, especially at lower traffic loads and larger packet sizes. When more packets have to be retransmitted, we can balance this effect by saving time through reducing the number of accesses to the channel. This explains why end-to-end latencies at higher loads, for packet sizes equal to 100Bytes, are lower than for the 32Bytes case. We have also considered a packet size of 320Bytes, which is not showed in the results to avoid crowding the figures with too many curves. The Sim320 results show a similar trend to that seen for Sim100. Having a packet size 10 times longer results in 10 times fewer control packets compared to Sim, achieving a throughput efficiency of 1. At the same time, the transmission delay for each data packet is ten times longer (32s for each data packet) and the resulting latency is therefore also much longer, especially for lower traffic loads (around 3.6 times longer at $\overline{\lambda} = 0.08$). The Sim320 latency gets closer to Sim when the traffic load increases (12% shorter for CSMA at $\overline{\lambda} = 0.213$, 8% shorter for DACAP at $\overline{\lambda} = 0.128$ and 57% longer for T-Lohi at $\overline{\lambda} = 0.213$).

The second question, concerning the importance of correctly accounting for actual hardware delays, is answered by comparing Simulated performance, with and without delay data. SimNoDelay indicates results of simulations in which we did not add the extra delays imposed by modem-dependent processing and information exchanges to each packet transmission. Without considering modem and Gumstix additional delays, the time to transmit and receive a packet is much shorter than in real-life, thus resulting in reduced latencies and increasing predicted throughput over that experienced at sea. The gap between simulation and experimental data can be significant if these delays are not accounted for. For example, the DACAP and CSMA packet latency displayed by the SimNoDelay curves is as low as one third of the experimental result, while T-Lohi's latency when not accounting for modem-dependent delays is as low as half the experimental result.

Table III
STATIC MULTI-HOP SCENARIO (0.085 PPT)

Protocols	Kind	Th	L	D	T
CSMA	Sea	0.64	73.21	2.1	-
	Sim	0.89	62.19	1.55	-
	SimNoDelay	0.94	32.08	1.37	-
	Sim100	0.97	48.45	1.27	-
T-Lohi	Sea	0.56	38.72	1.15	1.55
	Sim	0.83	32.31	1	1.125
	SimNoDelay	0.89	30.23	1	1.11
	Sim100	0.91	42.02	1.01	1.08

We have performed simulations and at-sea experiments with a mix of static and mobile node experiments, using both single-hop and multi-hop scenarios. Table III shows the result for the multi-hop scenario. In this case, with respect to the static scenario, the differences between Sea and Sim results have to be taken into account over the multiple traversed links. Differences in the results are mainly due to the higher number of attempts to advance packets over each traversed link we experienced when real acoustic devices are used. Throughput efficiency can be improved by reducing overhead and delays with the use of longer data packets. Again we can see how, ignoring additional delays which occur in real-life tests (SimNoDelay), simulation results differ substantially from real data.

Table IV
MOBILE SINGLE-HOP SCENARIO 1 (0.16 PPT)

Protocols	Kind	Th	L	D	T
CSMA	Sea	0.82	32.06	2.15	-
	Sim	0.92	19.05	1.50	-
	Sim2	0.85	26.75	1.72	-
	SimNoDelay	0.96	10.75	1.44	-
	Sim100	0.99	22.48	1.02	-
T-Lohi	Sea	0.78	21.43	1.09	1.22
	Sim	0.87	15.78	1	1.11
	Sim2	0.86	15.86	1	1.12
	SimNoDelay	0.95	12.61	1	1.02
	Sim100	0.97	18.34	1	1.05

Looking at Table IV, there is an interesting feature to point out in the mobile single-hop scenario 1 (the one involving 3 static nodes, the gateway buoy and the CRV Leonardo moving on the path A-B-C-D). In this case, we can see a much larger gap between Sim and Sea results, especially for CSMA. The reason is that, due to the noise produced by the engines on the CRV Leonardo, this mobile node was able to correctly hear only packets coming from node 2, which is at shallower depth with respect to nodes 1 and 3. The modem on CRV Leonardo also misses almost all the ACK packets coming from the sink node. T-Lohi is less affected by this behavior because it does not require an ACK or re-transmissions. When the sink, which can hear the modem on the CRV Leonardo, receives the data packet the transmission is completed and the CRV Leonardo node can then move on to transmit the next data packet, if any. For CSMA, although the sink correctly receives the data packet transmitted by CRV Leonardo, the ACK is not correctly received and CRV Leonardo kept transmitting the same data packet until the maximum number of transmissions is reached. This is an example in which a clear asymmetry in the connectivity between two nodes has a profound impact on protocol performance. The simulated results do not capture these features and therefore the gap between experimental at-sea traces and simulation results becomes quite large. We have therefore re-run simulations with a probability to correctly transmit packets over each link which reflects that measured in our experiments. The results of these simulations are labeled Sim2, and appear quite close to the experimental data.

A combination of accurate simulation and experimental validation therefore appears the most promising approach for assessing the performance and effectiveness of protocols proposed for UWSNs.

Table V
MOBILE SINGLE-HOP SCENARIO 2 (0.16 PPT)

Protocols	Kind	Th	L	D	T	R
CSMA	Sea	0.80	31.89	2.09	-	-
	Sim	0.92	18.98	1.53	-	-
	SimNoDelay	0.95	10.06	1.50	-	-
	Sim100	0.98	18.77	1.32	-	-
T-Lohi	Sea	0.72	14.33	1.11	1.52	-
	Sim	0.86	14.98	1	1.08	-
	SimNoDelay	0.94	12.70	1	1.06	-
	Sim100	0.98	18.38	1	1.03	-
Dacap	Sea	0.74	88.08	1.23	-	3.78
	Sim	0.90	48.79	1.1	-	1.36
	SimNoDelay	0.98	20.96	1.09	-	1.16
	Sim100	1.0	30.06	1.08	-	1.18

Table V presents the results for the second mobile scenario. The higher dynamics related to the presence of mobile nodes pose significant challenges to correctly receive packets and are harder to capture through simulations. The reason for the different performance between simulation and experiment is partially explained by significant differences in the number of attempts to access the channel for data, RTS and Tone packets. This in turn results in longer delays, a higher number of packets in the channel at a given time and therefore a lower throughput efficiency.

IV. CONCLUSIONS

We have performed a comparative performance evaluation of three among the most renown MAC schemes for UWSN: DACAP, T-Lohi and CSMA. The performance of these three different protocols have been evaluated from traces recorded during extensive tests off Pianosa island during ACommsNet10. Subsequently, we simulated the exact same scenarios (node locations, node roles, etc.) in order to compare simulation and sea trial results. Looking at the results (Section III), our comparison between simulation results and experimental data at first showed significant gaps when the acoustic channel model insufficiently captured the channel physics and, more importantly, when the detailed operations, limitations and delays of the acoustic modem were not included. Once the channel model was refined and all the overhead and delays related to the different components involved during the experiments were included, the gap between experimental at-sea traces and simulation results reduced significantly. While our results show that no state of the art simulator is able to capture all the effects that might impair UWSN performance, they also clearly demonstrate that the choice of specific devices for experimental activities can significantly affect experienced performance. To provide evidence of this, we varied the packet size over a range appropriate for the FSK Micro-Modem). Specifically, we investigated the impact on performance of different possible packet sizes. The results show that larger packet sizes can lead to significantly better system performance in terms of throughput efficiency, at a cost of increased packet latency, especially for low traffic loads. We conclude that simulations can be used to accurately predict real-life performance at sea, provided care is taken to capture all the important acoustic propagation physics and the physical attributes of the hardware used in the experiment. Ignoring the actual delays and overhead related to the specific acoustic modem used for the experiments and ignoring the effect they have on the protocol stack setting can result in bad protocol settings and selections, degrading the network performance. A combination of accurate simulation and experimental validation therefore appears the most promising approach for assessing the performance and effectiveness of protocols proposed for UWSNs. We have also discussed many practical effects that we have experienced and that cannot be easily captured by simulators. We believe that these are primarily responsible for the remaining performance gap. Moreover, we show how acoustic modem operations and limitations can strongly affect at-sea performance and how overcoming some of these limitations can strongly improve the network performance in terms of throughput efficiency and packet latency.

ACKNOWLEDGMENTS

The authors would like to thank the Parco Nazionale Archipelago Toscano, the Engineering support and the Ship Management Office at NURC, who made this experiment possible. Special thanks to "il governatore" for always being on hand, with radio and at least two handphones, to immediately solve any problem arising. This work was supported in part by the NURC Consolidated Programme of Work and the EU FP 7 STREP project CLAM "CoLlAborative EMbedded Networks for Submarine Surveillance" .

REFERENCES

[1] C. Petrioli, R. Petroccia, and M. Stojanovic, "A comparative performance evaluation of MAC protocols for underwater sensor networks." in *Proceedings of MTS/IEEE OCEANS 2008*, Quebec City, Quebec, Canada, September 15–18 2008.

[2] P. Casari, B. Tomasi, and M. Zorzi, "A comparison between the tone-lohi and slotted fama mac protocols for underwater networks." in *Proceedings of MTS/IEEE OCEANS 2008*, Quebec City, Quebec, Canada, September 15–18 2008.

[3] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "ALOHA-based MAC protocols with collision avoidance for underwater acoustic networks," in *Proceedings of IEEE INFOCOM 2007, the 26th IEEE International Conference on Computer Communications*, Anchorage , Alaska , USA, May 6–12 2007.

[4] S. Basagni, C. Petrioli, R. Petroccia, and M. Stojanovic, "Choosing the packet size in multi-hop underwater networks," in *Proceedings of IEEE OCEANS 2010*, Sydney, Australia, May, 24–27 2010, pp. 1–9.

[5] B. Peleato and M. Stojanovic, "Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks." *IEEE Communications Letters*, vol. 11, no. 12, pp. 1025–1027, December 2007.

[6] A. Syed, W. Ye, and J. Heidemann, "Understanding spatio-temporal uncertainty in medium access with ALOHA protocols," in *Proceedings of the third ACM International Workshop on UnderWater Networks (WUWNet '07)*, Montréal, Quebec, Canada, September 14 2007, pp. 41–48.

[7] J. Yackoski and C. Shen, "UW-FLASHR: Achieving high channel utilization in a time-based acoustic MAC protocol," in *Proceedings of the third ACM International Workshop on UnderWater Networks (WUWNet '08)*, San Francisco, California, USA, September 15 2008, pp. 59–66.

[8] M. Stojanovic and L. Freitag, "Wideband underwater acoustic CDMA: adaptive multichannel receiver design." in *Proceedings of MTS/IEEE OCEANS 2005*, vol. 2, Washington, DC, USA, September 17–23 2005, pp. 1508–1513.

[9] R. Diamant and L. Lampe, "A hybrid spatial reuse mac protocol for ad-hoc underwater acoustic communication networks," in *Communications Workshops (ICC), 2010 IEEE International Conference on*, Cape Town, South Africa, 23–27 May 2010, pp. 1–5.

[10] 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/micromodem/, 2005.

[11] A. S. Tanenbaum and D. Wetherall, *Computer Networks*, 5th ed. Upper Saddle River, NJ: Prentice Hall PTR, 2010.

[12] A. Syed, W. Ye, and J. Heidemann, "Comparison and evaluation of the T-Lohi MAC for underwater acoustic sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 9, pp. 1731–1743, December 2008.

[13] C. Petrioli, R. Petroccia, J. Shusta, and L. Freitag, "From underwater simulation to at-sea testing using the ns-2 network simulator," in *Proceedings of IEEE OCEANS 2011*, Santander, Spain, June, 6–9 2011.

[14] N. Baldo, F. Maguolo, M. Miozzo, M. Rossi, and M. Zorzi, "ns2-MIRACLE: A modular framework for multi-technology and cross-layer support in network simulator 2," in *Proceedings of the 2nd International Conference on Performance Evaluation Methodologies and Tools, ValueTools 2007*, Nantes, France, October 23–25 2007, pp. 1–8.

[15] M. Porter *et al.*, "Bellhop code." [Online]. Available: http://oalib.hlsresearch.com/Rays/index.html

[16] F. Guerra, P. Casari, and M. Zorzi, "World ocean simulation system (woss): a simulation tool for underwater networks with realistic propagation modeling," in *Proceedings of the Fourth ACM International Workshop on UnderWater Networks*, ser. WUWNet '09, Berkeley, California, USA, 3 November 2009, pp. 1–8.

[17] "Gumstix inc." [Online]. Available: http://www.gumstix.com

[18] Graaltech, "Folaga." [Online]. Available: http://www.graaltech.it/en/index.php