Performance Evaluation of Random and Handshake-Based Channel Access in Collaborative Mobile Underwater Networks

Federico Guerra*, Paolo Casari*, Alessandro Berni‡, John Potter‡, Michele Zorzi*

*Department of Information Engineering, University of Padova, Italy — {fguerra, casarip, zorzi}@dei.unipd.it

‡NATO Undersea Research Centre (NURC), La Spezia, Italy — {berni, potter}@nurc.nato.int

***Abstract*—In this paper, we compare communication schemes in scenarios of interest for underwater networks where multiple nodes collaborate towards the accomplishment of a target detection and tracking task. We consider two specific cases: in Scenario 1, a mobile Autonomous Underwater Vehicle (AUV) i) collects data from a set of fixed sensors deployed on the seafloor or ii) transmits data to the same nodes. In Scenario 2, two AUVs are placed at opposite sides of an area to be patrolled, and move in the same direction. Some low-complexity, relay-only AUVs keep moving in between edge AUVs in order to support networked data exchange between them.**

We compare the performance of random as well as handshake-based communications by means of the ALOHA and the DACAP protocols, respectively, in both scenarios. Simulations are performed in a realistic environment, where environmental parameters are extracted from ocean databases and fed to the Bellhop channel simulator through the WOSS framework which seamlessly interfaces the nsMIRACLE network simulator with the more accurate channel representation provided by Bellhop.

Results show that while base protocol configurations offer reasonable performance only in scenarios with limited traffic, simple improvements such as back-to-back packet transmission and power control provide significant performance improvements.

I. Context and Scenario Description

Cooperative sensing, data acquisition, exchange and processing are part of many underwater networking applications. Collaborative, autonomous target detection and tracking (e.g., for threat assessment assistance) makes no exception to this point. These tasks usually entail monitoring, patrol, or search of a possibly wide area for specific targets; specific assets will then report back a picture of the situation on the field to a Command and Control (CC) center, which will take and direct appropriate actions or countermeasures.

The future vision for collaborative detection and tracking (CDT) systems entails the use of several autonomous assets, with possibly limited capabilities but also much cheaper than any manned vessels. Some of these assets may just be sensing devices; others may have the ability to move and carry more complex sensors (such as an underwater camera); some devices may be moored, others anchored to the sea bottom, others mobile. In any event, the nodes will be networked, and leverage on underwater communications to exchange data, possibly move to gather information from fixed nodes which would be commonly out of range, and collaboratively conjure up a picture of ongoing events. In this light, it also makes sense that some nodes be deployed to work only as communication relays, or as gateways between the acoustic and radio communication worlds. While this vision becomes increasingly close to be realized, many works on CDT explore how to cooperatively perform tracking missions so that some performance objective is reached. For example [1] employs an optimal control framework to effectively maximize the coverage of a certain phenomenon as provided by a swarm of autonomous vehicles; in [2] the author also considers the problem of tracking and employ to this end a localization algorithm based on progressive refinement of regions: each region is reduced in size at every measurement received, and increased in size as no new measurement becomes available for a certain time; the focus of [3] is on the minimization of formation errors for underwater vehicles moving in formation as they track and follow an event; the authors in [4] focus on exchanging data between two vehicles equipped with passive sensors and running the MOOS [5] platform while following a target whose trajectory is unknown.

In this paper, we take one step back and consider instead the problem of efficiently exchanging data between nodes in a CDT scenario. We consider two different approaches, which have been shown to be characterized by different performance depending on the traffic generation pattern of the network [6]: the first is ALOHA [7], which besides being simple is also straightforward to implement within multi-purpose autonomous systems [5] where communication is only one of the many tasks which have to be coordinated on a node; the second protocol is the Distance-Aware Collision Avoidance Protocol (DACAP) [8], which offers protection from interference to ongoing transmissions by means of a more involved signaling phase: instead of allowing for a completely random access, this phase puts to silence potentially harmful interference within an area of prescribed size, and adopts further measures to prevent transmissions which would certainly collide. We carry out the analysis in two different scenarios: the first features fixed nodes and one mobile node; the second focuses on a fully mobile network.

For the purposes of the present work, we will proceed by means of simulations using the World Ocean Simulation System (WOSS) [6], the framework we developed to interface the wireless network simulator nsMIRACLE [9] with the acoustic propagation simulator Bellhop [10]. In particular, Bellhop implements a ray model which is employed to reach an approximate solution to the propagation equations: to do this, it requires boundary conditions represented by environmental factors such as the sound speed profile (SSP), the bathymetry and surface profiles, and the type of bottom sediments in the area where the transmitter and receiver nodes are located. WOSS automates the extraction of this data from free oceanographic databases (the World Ocean Database [11] for SSPs, the General Bathymetric Chart of the Oceans [12] for bathymetry samples with an angular spacing of 30 seconds of arc, the National Geophysical Data Center's Deck41 database [13] for the type of bottom sediments), starting from the geographic location of the transmitter and receiver. This way, the user only has to specify where the nodes are located at the moment of the transmission in order to get a channel snapshot. Since nsMIRACLE, and likewise other network simulators, commonly handle node positions both in case of static and mobile networks, WOSS provides a good means of connecting those simulators to a more realistic physical layer than provided, e.g., by empirical formulas such as those found in [14].

The comparisons carried out in this paper regard two different scenarios relevant to CDT operations, and involving fixed and mobile nodes. These scenarios will be discussed in the next section. Section III will shortly describe the details of the ALOHA and DACAP protocols considered in this study, highlighting expected pros and cons of either approach. Section IV will describe the simulation results both in the absence and in the presence of back-to-back packet transmissions and power control, before concluding remarks are drawn in Section V.

II. Scenarios

The scenarios described hereafter have been designed to reproduce typical operations to be carried out in the CTD context, namely, data collection from fixed devices (that are out-of-range of data collection stations, either ashore or afloat), and mobile-to-mobile communication through a completely mobile network. All scenarios have been simulated in a reference environment for sea trials, such as the test site off the eastern coast of the Pianosa island, Italy, used by the NATO Undersea Research Centre (NURC) for the experiments of the SubNet 2009 trials [15]. The considered environment is therefore shallow water. In particular, for Scenario 1 the bathymetry has been simulated (i.e., not taken from databases) and given a "Pianosa-like" mildly sloping-down profile (roughly from a depth of $50\,\mathrm{m}$ to $200\,\mathrm{m}$ in a $2\,\mathrm{km}$ range). For Scenario 2, instead, real bathymetry data was employed.

A. *Scenario 1: four fixed nodes and one AUV*

This scenario is used to test the time it takes a mobile Autonomous Underwater Vehicle (AUV) to i) collect data from a set of fixed sensors deployed on the seafloor and ii) transmit data to the same nodes. For a pictorial description of the scenario, we refer to Figure 1. The scenario entails:

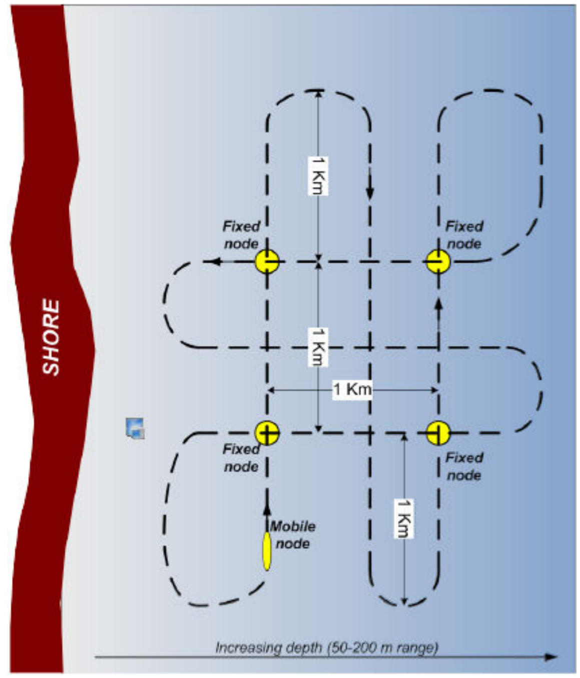

Figure 1. Scheme of scenario 1: 4 fixed, bottom mounted nodes and 1 AUV patrolling the area following a closed-loop trajectory.

- 1 mobile underwater node (M1) equipped with the WHOI micromodem (MM) [16] operated in the C band (roughly $23\,\mathrm{kHz}$ to $27.5\,\mathrm{kHz}$) and corresponding co-processing boards for high-speed underwater communications up to $4800\,\mathrm{bps}$;
- 4 fixed nodes deployed on a 2×2 square grid with nearest neighbors $1\,\mathrm{km}$ apart, equipped with the same communication hardware as the AUV.

The AUV patrols the area at a speed of 1 to $4\,\mathrm{knots}$ by describing a back and forth trajectory which leads it on top of each node in sequence, by first moving in the East-West direction, and then in the North-South direction; the trajectory is a closed loop, is about $16\,\mathrm{km}$ long, and is completed in roughly 2 hours and 15 minutes if the AUV cruises at a full speed of $4\,\mathrm{knots}$ (about $7.4\,\mathrm{km/h}$). In any event, the duration of the mission is set to 9 hours, so that all traffic can be delivered and the whole path of the AUV can be repeated more than once for any considered speed.

We distinguish between two different traffic patterns: an AUV-to-nodes and a nodes-to-AUV pattern. The first represents a simple test of communications performance in a very baseline case using the protocols and schemes considered in this paper; the second pattern is used to test multiple access performance in a simple data gathering network with a limited number of nodes. For the AUV-to-nodes traffic, the AUV generates an amount of $5\,\mathrm{KiB}$ of data every $10\,\mathrm{min}$ ($1\,\mathrm{KiB} = 1024\,\mathrm{Bytes}$): this data must be sent to all fixed nodes. For the nodes-to-AUV traffic, each fixed node generates only one $50\,\mathrm{KiB}$-long data message which must be uploaded to the AUV. In either case, the fixed nodes do not know the trajectory of the AUV, which acts autonomously based on its own mission arrangements.

B. Scenario 2: fully mobile network of AUVs

Unlike in scenario 1, here all assets are mobile. However, some assets are assumed to need a networked data exchange which should be carried out through a network of mobile nodes. These nodes can be, e.g., simpler or cheaper AUVs swarming in the area between the two more expensive ones. With reference to Fig. 2, the network is therefore configured as follows:

- 2 high-value AUVs positioned at the edges of the network, approximately $10\,\mathrm{km}$ apart
- A swarm of 4 AUVs that act as relays from in between the edge nodes

In this scenario, the main requirement is to investigate the feasibility of two-way exchange (every minute) of data files that are 1 to $5\,\mathrm{KiB}$ in size between the high-value AUVs at the edges. Such a requirement entails the configuration of end-to-end routing between the assets, as one-hop connectivity will be likely unavailable between two communicating parties placed at such distance. To this end, the movement pattern of the edge AUVs is fully deterministic, as they start from fixed coordinates (see Fig. **??** for a geographic reference picture) and move due south at a constant speed of $2\,\mathrm{knots}$, roughly $1\,\mathrm{m/s}$. On the contrary, some variability occurs in the trajectory of the other AUVs, in order to simulate some local, autonomous behavior. In more detail, these AUVs move on average at a speed v_1 of $2\,\mathrm{knots}$ in the same direction of the edge AUVs, but their velocity has two additional components: the first, v_2, is oriented in the horizontal (East-West) direction, and simulates lateral displacements; the second component, v_3, is oriented in the same direction as v_1, and simulates temporary increases or decreases with respect to the otherwise fixed advancement speed of $2\,\mathrm{knots}$. The values of v_2 and v_3 are initially 0, and are set to a random value periodically, as the speed vector of the node is updated. Specifically, v_3 is set to a random value within the interval $[-0.25, +0.25]$, whereas v_2 is derived from v_1 and v_3, while ensuring that the absolute value of the overall speed vector is not greater than $4\,\mathrm{knots}$: this is done by choosing a random value in the interval $[-v_2^{\max}, +v_2^{\max}]$, where $v_2^{\max} = \frac{1}{2}\sqrt{4 - v_1^2 - v_2^2}$ (all speeds are expressed in knots). In any event, the duration of the mission does not exceed the time required for the edge nodes to cover a length of $10\,\mathrm{km}$.

C. Additional details on simulation scenarios

Transmissions are assumed to take place using WHOI MM-like hardware: in this view, the fixed transmission bit rate is $4800\,\mathrm{bps}$. Data has been fragmented in packets of fixed size of $512\,\mathrm{bytes}$. The WHOI MM does not currently provide support for power control. In order to explicitly simulate available-in-hardware but unfavorable node configurations, transmissions will be first assumed to take place at full power; in addition we will initially assume to transmit one packet per channel access. However, we wish to simulate more favorable conditions even though they would not be currently implementable in the WHOI MM: in this light, we will consider a power control case where in addition packets are transmitted back-to-back in groups of given length, in order to increase the channel occupancy once channel access is gained and to improve the efficiency of the channel access protocols.

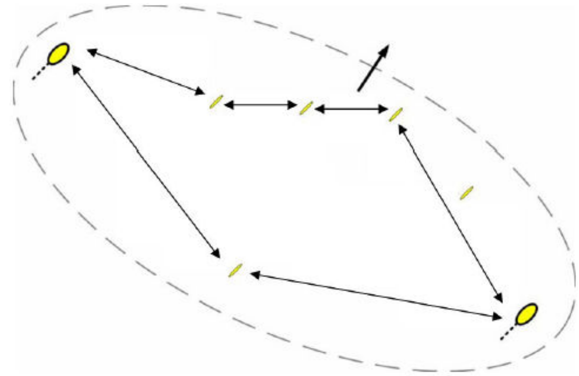

Figure 2. Scheme of Scenario 2: a fully mobile network with two external, more valuable nodes communicating through a network of cheaper mobile nodes in between.

As to Scenario 2, we note that the large distance separating edge nodes requires a routing protocol to be implemented. In this paper, we consider a form of geographic routing. In more detail, every node with a packet to transmit seeks relays within a cone of 90° of aperture, pointed towards the likely position of the destination after the expected travel time of the signal. In other words, the transmitters are assumed to know what the trajectory of the edge AUVs will be, and calculate the position they will be found at after the time it would take to complete a transmission. Among different relay choices, the node within the cone which is closest to a predetermined distance of $6.5\,\mathrm{km}$ from the transmitter is chosen.

III. Simulated MAC protocols

For the evaluation of the scenarios considered in the previous section, we employ two different protocols, which are representative of different tradeoffs between handshaking complexity and protection from channel access errors and collisions: ALOHA [7] and DACAP [8].

ALOHA is the simplest access protocol for wireless networks, as it prescribes that a transmission can take place whenever there is a packet to transmit. It is very inefficient at high traffic regimes, but works well for low traffic. In this paper we actually consider (and interchangeably name) CSMA-ALOHA, whereby a short channel sensing is performed before transmitting, so that the node backs off if the channel is busy and can transmit again as soon as it returns clear. This avoids two types of straightforward collisions: transmit collision, whereby a receiver starts transmitting while reception of a packet for itself is taking place, and receive collision, whereby a node transmits while receiving a packet meant for another node, thus creating harmful interference to the intended receiver of the packet. We will consider both acknowledged and non-acknowledged receptions, whereby an ACK message is or is not sent back to the transmitter to confirm that a correct packet has been received.

The second protocol we consider here, namely the Distance-Aware Collision Avoidance Protocol (DACAP) [3] is based on the limitation and avoidance of collisions through a 3- or 4-way handshaking scheme (depending on the absence or presence of ACKs, respectively). The protocol is specified as follows. The transmitter notifies its will to access the channel through a Request-To Send (RTS) message, and the receiver replies with a Clear-To-Send (CTS) upon receiving the RTS. At this point, the transmitter enters a waiting state, where four different events may occur: 1) the channel is clear; 2) the receiver, after transmitting the CTS, overhears a packet threatening its pending reception; 3) the transmitter, during the defer time, overhears a packet meant for some other node or receives a special warning message from its partner; 4) the CTS is lost. In case 1) the data transmission takes place right after the end of the defer interval; in case 2) the receiver sends a short special warning message to the transmitter: if received on time, this message prevents sending data which would surely collide; in case 3) the sender automatically defers its transmission to avoid collisions; in case 4) the sender backs off and retries at a later time. The length of the idle period after the reception of the CTS has been designed so that only the signals from neighbors within a certain area are considered harmful, thereby bounding the average the Signal-to-Interference-and-Noise ratio (SINR) to a value high enough to carry out the transmission. To achieve a trade-off that maximizes the throughput of a given network, a minimum hand-shake length $t_{\min}$ is predefined for all the nodes. For a network in which most links are close to the transmission range, $t_{\min}$ needs to be nearly twice the maximum propagation delay. When some links are shorter, it can be reduced. For example, a natural choice in Scenario 2, is to set $t_{\min}$ such that any link within the distance of $6.5\,\mathrm{km}$ (at which a relay is looked for) is safe from interference, at least if no collisions involve signaling messages, or signaling and data messages. While ACKs may be added to the handshake above in order to confirm correct reception, the good performance of DACAP in avoiding collisions between terminals makes it unlikely that ACKs are actually required. This is confirmed by the results we will discuss later.

IV. Simulation Results

A. Scenario 1

Simulations carried out in Scenario 1 (Fig. 1) using the nodes-to-AUV traffic pattern expectedly show that ALOHA is more efficient, as it requires neither messaging overhead nor the corresponding time period to be allowed for signaling messages to travel back and forth between communicating parties. Part of this overhead is alleviated by the use of back-to-back packet transmissions: in fact, the transmitter makes more efficient use of the handshake, whose overhead is split among multiple packets.

In case the nodes have to upload data to the AUVs, a local congestion is created by multiple access attempts, the signals can potentially collide, and ALOHA tends to be less efficient than DACAP. This can be observed from the results depicted in Figures 3 and 4, showing the application throughput (in bps) and the transmission failure probability (the ratio of failed packet transmissions to all transmissions performed by all nodes) for ALOHA and DACAP, as a function of the AUV movement speed. In these and in the following figures, back-to-back packet transmissions are indicated using the acronym "B2B." [1] In order to make results comparable, we remark that throughput is averaged over the whole duration of the mission of the AUV, which may contain silence periods if the data transfer is completed before the end of the mission. Let us focus on the plain, no-ACK versions of both ALOHA and DACAP for the moment. From Fig. 3, we observe that repeated channel access attempts without regulation and subsequent collisions harm ALOHA, as they cause unsuccessful transmissions and correspondingly decrease throughput. By way of contrast, DACAP's heavier handshaking procedure is effective here, as it allows channel access to only one node at a time (using the discussed parameters); the net effect is a lower failure probability (always lower than 7%), and consequently a higher throughput of up to almost $12\,\mathrm{bps}$.

In order to understand whether error control makes ALOHA more efficient in this scenario, we included in both figures an ACK version of ALOHA, where a Stop-and-Wait Automatic Repeat reQuest (ARQ) policy is employed to recover errors. However, in congested scenarios as the one considered here, ACK messages do allow greater throughput, but also represent a further source of collision, delay and overhead, which reduce the benefit of error recovery. For this reason, the throughput of ALOHA with ACKs is higher than for the version without ACKs, but the probability that a transmitted packet is lost is still very high (Fig. 4), and in any event much higher than DACAP's in the no-ACKs version.

A much higher performance improvement is provided, to both ALOHA and DACAP, by the use of back-to-back packet transmissions. In this case, a total of 8 packets are grouped into a super-packet at the application level and transmitted over the channel. When the super-packet is received, each packet is separately checked for errors and passed to the application. We recall that this strategy leads to fundamental benefits in terms of channel access efficacy (longer occupancy makes both CSMA-ALOHA and DACAP more likely to be aware that the channel is busy), and to better use of overhead signaling for DACAP. In particular, the latter protocol features a handshake phase which may be used to perform power control by tuning down transmit power whenever the receiver of the message notifies in its CTS that its perceived SNR is higher than a prescribed threshold, fixed here to $20\,\mathrm{dB}$. The drawback of longer transmissions, i.e., higher vulnerability to collisions

[1] As per the discussion in subsection IV-A, the B2B DACAP version also includes power control. We note that we also tried DACAP versions both using only power control and using only B2B transmissions, and B2B offers the most significant performance improvements with respect to power control.

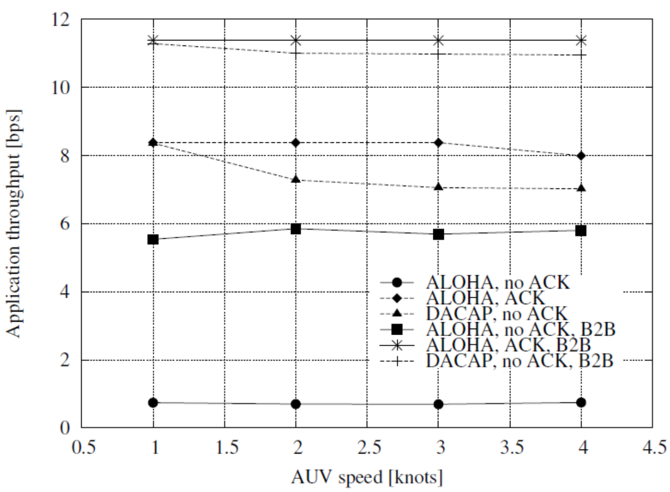

Figure 3. Application throughput in bps for ALOHA (with and without ACK) and DACAP (without ACK) in Scenario 1, traffic pattern 2, as a function of AUV movement speed.

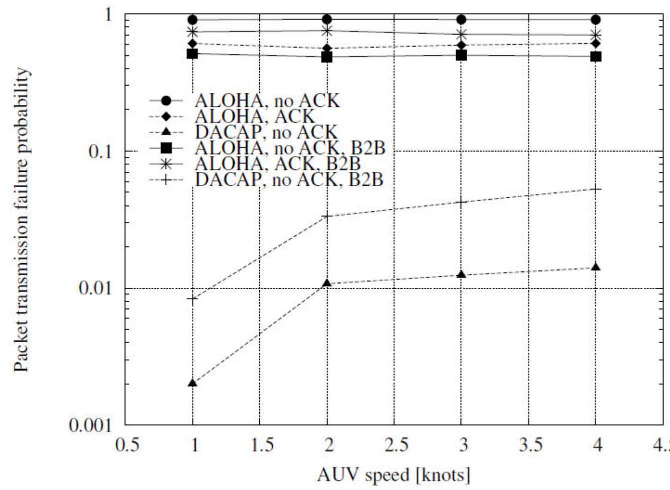

Figure 4. Packet transmission failure probability for ALOHA (with and without ACK) and DACAP (without ACK) in Scenario 1, traffic pattern 2, as a function of AUV movement speed.

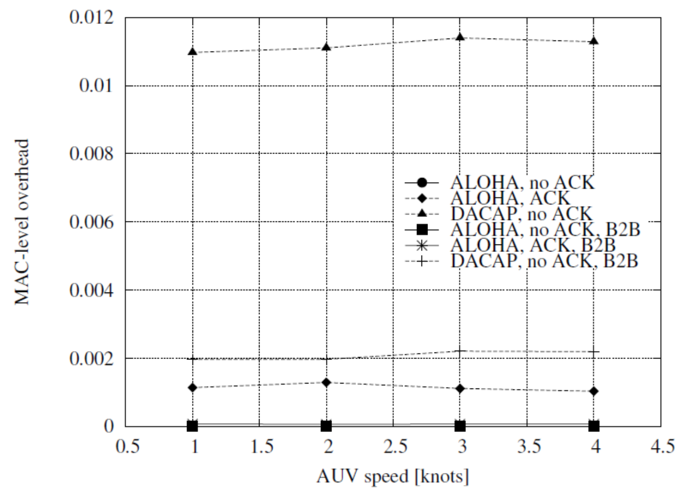

Figure 5. MAC-level overhead for ALOHA (with and without ACK) and DACAP (without ACK) in Scenario 1, traffic pattern 2, as a function of AUV movement speed.

from signaling packets, is also compensated for by the fact that signaling packets are short, and can therefore affect only a portion of the packet train (i.e., at most two consecutive packets). The net effect seen in Fig. 3 is better throughput performance for ALOHA. In relative terms, the no-ACK version achieves greater improvements, but in absolute terms the best performance is achieved by the ACK version, where the use of back-to-back transmissions increases throughput from $8.4\,\mathrm{bps}$ to $11.5\,\mathrm{bps}$. DACAP also experiences higher throughput, even though its probability of packet reception failure is slightly higher due to more likely collisions from signaling messages. These results are in line with those in similar performance studies such as [17], which has been carried out considering a fixed network, where packets are fragmented in smaller chunks rather than grouped into super-packets. A further difference between this paper and [17] is the use of a more realistic physical layer obtained by simulating acoustic propagation through Bellhop [10].

The further benefits yielded by back-to-back packet transmissions is shown in Fig. 5, which depicts MAC-level overhead (defined as the ratio of signaling bits transmitted to all correctly received data bits) as a function of the AUV movement speed. In this case, the base version of DACAP incurs the highest overhead: back-to-back transmissions of 8 packets per channel access help reduce this (by less than a factor of 8, however, as only correctly received packets are considered in the denominator of the overhead formula).

Conclusions on scenario 1 In accordance to results in similar performance comparisons [6], fully random access does not appear to be a viable option here, due to the very high number of collisions caused by concurrent access. Adding ACK messages for error control does yield significant improvements, insofar as packets are grouped so as to improve the efficiency of channel access efforts. In case no ACK messages are to be implemented, DACAP represents a good solution, as it yields reliability through a better administration of multiple access; in addition, back-to-back packet transmissions help reduce the 3-way handshaking overhead.

B. Scenario 2

We will now switch to scenario 2 (Figs. 2 and **??**). Given the more erratic nature of traffic in this kind of network, we will consider the versions without ACK of both ALOHA and DACAP, and compare single packet transmissions against transmissions of multiple packets per channel access. Fig. 6 details this comparison by showing the application-level throughput as a function of the traffic generation rate in the network (we recall that traffic is generated only by the two AUVs positioned at the opposite sides of the network). We observe that the single-packet versions of the two protocols perform almost equivalently, suggesting that the network can support

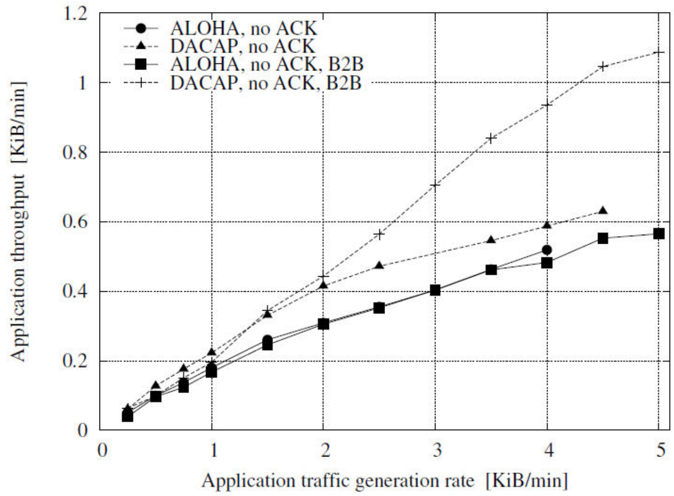

Figure 6. Application-level throughput for ALOHA and DACAP, in Scenario 2, as a function of the traffic generation rate. Both single-packet and multiple back-to-back packet transmissions are shown.

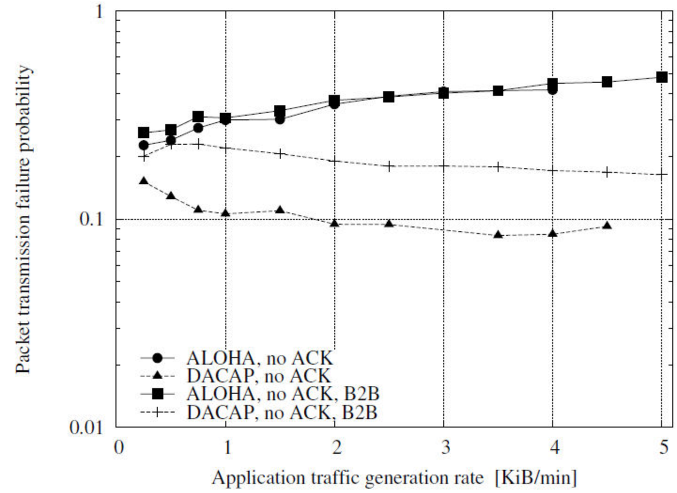

Figure 7. Packet transmission failure probability for ALOHA and DACAP, in Scenario 2, as a function of the traffic generation rate. Both single-packet and multiple back-to-back packet transmissions are shown.

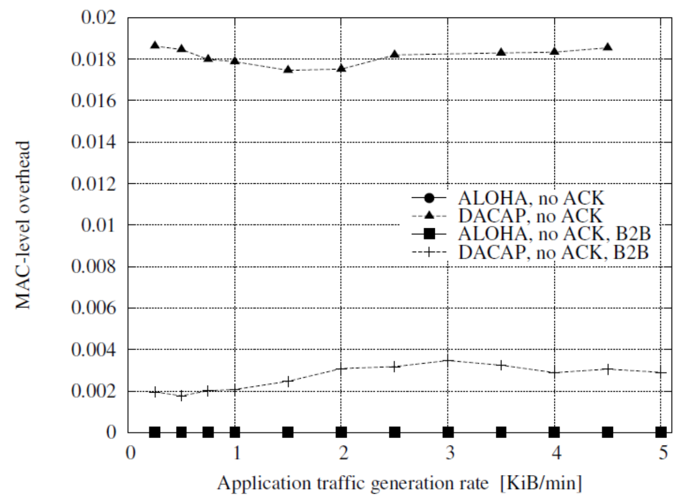

Figure 8. MAC-level overhead for ALOHA and DACAP, in Scenario 2, as a function of the traffic generation rate. Both single-packet and multiple back-to-back packet transmissions are shown.

the traffic demand. In particular, throughput increases quite smoothly though in a slightly sub-linear way for increasing traffic generation rate. However, the actual throughput value is much less than the offered traffic, because of delays generated by multihop routing procedures throughout the area between the two edge AUVs. The ineffectiveness of single-packet transmissions is also highlighted by the better performance of the back-to-back transmission versions.

Figure 7 details the probability of failing a packet transmission. The results reflect the behavior of throughput in Fig. 6, as the single-packet versions tend to lose between 2 and 5 packets every 10 transmissions, whereas the back-to-back versions feature lower error probabilities, down to 0.1 for DACAP, which can keep interference more controlled than ALOHA. Interestingly, the better error protection performance of DACAP allows to fill the gap created by long propagation delays during the handshake phase, allowing to exploit both the higher protection (and lower error probability) yielded by handshakes and the better channel utilization towards a higher throughput.

As a final comparison, we show in Fig. 8 the overhead of the protocols, which is expectedly higher than in scenario 1, due to the greater amount of (multihop) traffic that has to be supported in order to make a packet correctly reach the receiver. For example, the amount of overhead required by DACAP in scenario 2 is roughly twice the overhead measured in scenario 1, showing that about two hops are required to cover the distance between edge AUVs.

Conclusions on scenario 2 While in this scenario the network is not congested, two-way traffic between edge AUVs may be impaired by cross-interference and collisions at intermediate nodes used as relays. Back-to-back packet transmissions relieve the stress of both protocols, and in particular improve DACAP's performance to overcome ALOHA's, which suffers from greater interference due to unregulated access. While further optimization may be achieved by fragmenting packets (even those transmitted back-to-back) so as to further reduce the data bits affected by collisions [17], the results in this paper show that both ALOHA's random access and DACAP's handshake-based access can benefit from back-to-back transmissions.

V. Conclusions

In this paper, we have considered two different scenarios of interest for CDT operations. The first scenario includes 4 fixed, bottom mounted nodes and one AUV, and two traffic options (where the AUV has to upload data to all nodes, and vice-versa). The second scenario considers a fully mobile network of high-value AUVs placed at the edge of a network of cheaper AUVs which have to relay data transfers between the edge AUVs. Two MAC protocols have been considered, namely CSMA-ALOHA (representative of random channel access) and DACAP (representative of handshake-based channel access). The performance of the protocols in terms of throughput, success ratio, and overhead has been evaluated, discussing the differences in terms of complexity and reliability provided by the two approaches. For either scenario, back-to-back packet transmissions have been shown to improve the communication performance.

Future work on this topic includes improving the performance of routing by exploiting the characteristics of the scenario, e.g., by allowing pipelining in scenario 2, whereby the intermediate relays are given priority in forwarding operations whenever they receive packets from the edge AUVs.

ACKNOWLEDGMENT

This work has been supported in part by the NATO Undersea Research Center under contracts no. 40800700 (ref. NURC-010-08) and 40900654.

REFERENCES

[1] K. Baumgartner, S. Ferrari, and A. Rao, "Optimal control of an underwater sensor network for cooperative target tracking," *IEEE J. Ocean. Eng.*, vol. 34, no. 4, pp. 678–697, Oct. 2009.

[2] R. D. Scott, "Cooperative tracking for persistent littoral undersea surveillance," Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, Jun. 2007.

[3] E. A. Fiorelli, "Cooperative vehicle control, feature tracking and ocean sampling," Ph.D. dissertation, Princeton University, Princeton, NJ, Nov. 2005.

[4] D. Hughes, S. Kemna, R. Been, and M. Hamilton, "Sensible Behaviour Strategies for AUVs in ASW Scenarios," in *Proc. of MTS/IEEE Oceans*, Sydney, Australia, May 2010.

[5] M. R. Benjamin, J. J. Leonard, H. Schmidt, and P. M. Newman, "MOOS-IvP: An architecture and set of software modules for operating unmanned vehicles," Massachussetts Institute of Technology, Cambridge, MA, Tech. Rep. MIT-CSAIL-TR-2009-028, 2009.

[6] F. Guerra, P. Casari, and M. Zorzi, "World Ocean Simulation System (WOSS): a simulation tool for underwater networks with realistic propagation modeling," in *Proc. of ACM WUWNet 2009*, Berkeley, CA, Nov. 2009.

[7] L. G. Roberts, "ALOHA packet system with and without slots and capture," *ACM SigComm Computer Communication Review*, vol. 5, no. 2, pp. 28–42, 1975.

[8] B. Peleato and M. Stojanovic, "Distance aware collision avoidance protocol for ad hoc underwater acoustic sensor networks," *IEEE Commun. Lett.*, vol. 11, no. 12, pp. 1025–1027, Dec. 2007.

[9] N. Baldo, M. Miozzo, F. Guerra, M. Rossi, and M. Zorzi, "Miracle: the multi-interface cross-layer extension of ns2," *EURASIP J. of Wireless Commun. and Networking*, 2010. [Online]. Available: http://www.hindawi.com/journals/wcn/aip.761792.html

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

[11] "World ocean atlas." [Online]. Available: www.nodc.noaa.gov/OC5/WOA05/pr_woa05.html

[12] "General bathymetric chart of the oceans." [Online]. Available: www.gebco.net

[13] "National geophysical data center, seafloor surficial sediment descriptions." [Online]. Available: http://www.ngdc.noaa.gov/mgg/geology/deck41.html

[14] R. Urick, *Principles of Underwater Sound*. New York: McGraw-Hill, 1983.

[15] B. Tomasi, G. Zappa, K. McCoy, P. Casari, and M. Zorzi, "Experimental study of the space-time properties of acoustic channels for underwater communications," in *Proc. IEEE/OES Oceans*, Sydney, Australia, May 2010.

[16] 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," 2005. [Online]. Available: http://www.whoi.edu

[17] R. Petroccia, M. Stojanovic, S. Basagni, and C. Petrioli, "Optimizing network performance through packet fragmentation in multi-hop underwater communications," in *Proc. of IEEE Oceans*, Sydney, Australia, May 2010.