

Distributed Medium Access Control Strategies for MIMO Underwater Acoustic Networking

Li-Chung Kuo, *Student Member, IEEE*, and Tommaso Melodia, *Member, IEEE*

Abstract—The requirements of multimedia underwater monitoring applications with heterogeneous traffic demands in terms of bandwidth and end-to-end reliability are considered in this article. To address these requirements, a new medium access control protocol named UMIMO-MAC is proposed. UMIMO-MAC is designed to i) adaptively leverage the tradeoff between multiplexing and diversity gain according to channel conditions and application requirements, ii) select suitable transmit power to reduce energy consumption, and iii) efficiently exploit the UW channel, minimizing the impact of the long propagation delay on the channel utilization efficiency.

To achieve the objectives above, UMIMO-MAC is based on a two-way handshake protocol. Multiple access by simultaneous and co-located transmissions is achieved by using different pseudo-orthogonal spreading codes. An algorithm is proposed that, in a cross-layer fashion, jointly selects optimal transmit power and transmission mode through the cooperation of transmitter and receiver to achieve the desired level of reliability and data rate according to application needs and channel condition. Extensive simulation results show that UMIMO-MAC increases network throughput, decreases channel access delay, and decrease energy consumption compared with existing MAC protocols for UW-ASNs.

Index Terms—Underwater acoustic sensor networks, Medium access control, Multiple input multiple output.

I. INTRODUCTION

THE last few years are seeing a surge in research on underwater acoustic sensor networks [2]–[6]. However, currently available underwater acoustic technology can support only low-data-rate and delay-tolerant applications. Current experimental point-to-point acoustic modems use signaling schemes that can achieve data rates lower than 20 kbit/s with a link distance of 1 km over horizontal links. Academic experimental research activities have demonstrated modems for low-cost, short range, and low data rate (1 kbit/s) sensor networks [7]. Data rates as high as 150 kbit/s have been reported, but only on very short-range (≈ 10 m) vertical links, which are unaffected by multipath [8]. Typical commercially available modems provide even lower data rate waveforms [9]–[11]. For example, the Woods Hole Oceanographic Institution (WHOI) micromodem [12], which can be used in small-scale academic testbeds, is based on a frequency hopping, frequency shift

keying (FH/FSK) or phase shift keying (PSK) waveforms that support data rates of 80 – 5400 bit/s.

Recently, the availability of inexpensive hardware such as CMOS cameras and microphones able to ubiquitously capture multimedia content from the environment is enabling the so-called wireless multimedia sensor networks technology [13], i.e., wireless systems designed to retrieve video and audio streams, still images, and scalar sensor data from the environment. Similarly, multimedia underwater sensor networks would conceivably enable new applications for underwater multimedia surveillance, undersea exploration, video-assisted navigation and environmental monitoring. However, these applications require much higher data rates than currently available with acoustic technology and more flexible protocol design to accommodate heterogeneous traffic demands in terms of bandwidth, delay, and end-to-end reliability. For example, state-of-the-art predictive encoding of QCIF underwater video sequences, which are characterized by low contrast, generates streams of size in the order of several tens of kbit/s (e.g., [14]). Data rates at least twice as high are needed to transmit video over multi-hop paths with half duplex modems. There is, therefore, an urgent need to develop high-data-rate underwater acoustic communication technologies to support delay-sensitive multimedia sensor network applications with high bandwidth demands. To accommodate such traffic demands, we propose to leverage the potential of *multiple-input-multiple-output* (MIMO) transmission techniques on acoustic links and to develop new cross-layer communication protocols to adaptively exploit the performance increase offered by MIMO links under the unique challenges posed by the underwater environment.

The MIMO transceiver technology has attracted considerable attention in radio-frequency (RF) communications since the early 2000's [15], [16]. Instead of mitigating the impact of multipath fading, MIMO systems are able to exploit rich scattering and multipath fading to provide higher spectral efficiencies without increasing power and bandwidth. Hence, MIMO technology has the potential to take advantage of the rich scattering and multipath of the underwater acoustic environment to increase data transmission rates and improve link reliability. This idea is also being recognized by the underwater acoustic communication community in recent years. In [17]–[22], the feasibility of MIMO systems and related coding and modulation was tested for underwater acoustic communications and significant performance improvement was demonstrated compared with the conventional SISO system architecture. Experimental studies on underwater MIMO transmission techniques include [17], [18], [20], [23]–[25]

Manuscript received February 15, 2010; revised October 11, 2010 and March 15, 2011; accepted March 19, 2011. The associate editor coordinating the review of this paper and approving it for publication was J. Shea.

This work was supported by the National Science Foundation under grant CNS-1055945. A preliminary shorter version of this paper [1] appeared in the Proceedings of ACM MSWiM 2009.

The authors are with the Department of Electrical Engineering, State University of New York at Buffalo, Buffalo, NY, 14260 USA (e-mail: {lkuo2, melodia}@buffalo.edu).

Digital Object Identifier 10.1109/TWC.2011.061511.100233

and limited previous work on theoretical link performance optimization and analysis [22], [26]. The impact of the capabilities of MIMO links on higher layer protocols in underwater networks is however substantially unexplored. The objective of this paper is therefore to explore the capabilities of underwater MIMO links, and to leverage these at the medium access control layer. In particular,

- 1) We identify how the capabilities of MIMO links, in particular the tradeoff between transmission data rate and link error probability, impact MAC protocol design in underwater networks;
- 2) We develop a new MAC layer protocol called UMIMO-MAC that leverages MIMO capabilities to allow more flexible and efficient utilization of the underwater channel with respect to existing protocols. In particular, UMIMO-MAC is fully distributed and relies on lightweight message exchange. Moreover, UMIMO-MAC adapts its behavior to the condition of environmental noise, channel, and interference to maximize the network throughput or minimize the energy consumption, according to the Quality of Service (QoS) requirements of the traffic being transmitted;
- 3) We show how the principles that UMIMO-MAC is based upon can constitute basic building blocks to provide differentiated levels of QoS in underwater networks, and advance in the direction of *studying feasibility, limits, and solutions to transport multimedia traffic in underwater networks*.

The remainder of this paper is organized as follows. In Section II, we review recent literature on MAC layer protocols for underwater acoustic networks and on protocol design for wireless networks with MIMO links. In Section III, we introduce an underwater acoustic MIMO transceiver model. In Section IV, our proposed MAC protocol named UMIMO-MAC is introduced. In Section V, we assess the performance of the proposed solutions through simulation experiments. Finally, in Section VI, we draw the main conclusions.

II. RELATED WORK

Apart from studies concerned with acoustic communications at the physical layer [27], [28], recent research is concentrating on developing solutions at the medium access control (MAC) [29], [30] and network layers of the protocol stack. In [6], we proposed UW-MAC, a distributed MAC protocol tailored for UW-ASNs for which extensive carefully crafted simulations demonstrated that it achieves high network throughput, low channel access delay, and low energy consumption. UW-MAC is a transmitter-based code-division multiple access (CDMA) scheme that incorporates a novel closed-loop distributed algorithm to set the optimal transmit power and code length. In [31], a hybrid medium access control protocol is described, which includes a scheduled portion to eliminate collisions and a random access portion to adapt to changing channel conditions.

In [4], two distributed routing algorithms are introduced for delay-insensitive and delay-sensitive applications. The proposed routing solutions allow each node to select the optimal next hop, transmitting power, and strength of the forward error

correction algorithm, with the objective of minimizing the energy consumption. In [32], Harris and Zorzi study tradeoffs in the design of energy efficient routing protocols for underwater networks. In particular, an analysis is conducted that shows the strong dependence of the available bandwidth on the transmission distance, which is a peculiar characteristic of the underwater environment (see also Stojanovic [33]). Other significant recent studies consider delay-reliability tradeoff analysis (Zhang and Mitra [34]), the benefits achievable with cooperative communications (Mitra et al. [35]), and multi-path routing (Cui [36]).

Previous work has focused on developing MAC protocols for terrestrial wireless ad hoc networks with MIMO links. In particular, in [37], centralized and distributed MAC protocols for ad hoc networks with MIMO links called Stream-Controlled Medium Access (SCMA) are proposed. Throughput is increased by simultaneously transmitting multiple independent data streams on the same channel. In [38], a MIMO MAC protocol for ad hoc networks named MIMOMAN is proposed. The network throughput is enhanced by allowing simultaneous multiple communications at a higher data rate. However, the protocols in [37] and [38] do not consider the requirements of multimedia traffic, and consequently are not designed to leverage the multiplexing-diversity tradeoff typical of MIMO systems. In [39], a CDMA-based MAC protocol for wireless ad hoc networks is proposed, which deals with the near-far problem and interference margin. However, the considered near-far interference problem and receiver reservation do not consider the long propagation delay.

III. SYSTEM MODEL

A. Underwater Propagation Model

Underwater acoustic propagation [40] is substantially different from its RF counterparts [16]. Specifically, underwater acoustic communications are mainly influenced by *transmission loss, multipath, Doppler spread, and high propagation delay*. The transmission loss $TL(d, f)$ [dB] that a narrow-band acoustic signal at frequency f [KHz] experiences along a distance d [m] can be described by the Urlick model [40]:

$$TL(d, f) = \chi \cdot \text{Log}(d) + \alpha(f) \cdot d + A. \quad (1)$$

In (1), the first term account for *geometric spreading*. The second term accounts for *medium absorption*, where $\alpha(f)$ [dB/m] represents an absorption coefficient. The last term, expressed by the quantity A [dB], is the so-called *transmission anomaly*. More details can be found in [41].

B. Acoustic MIMO Transceiver Model

We consider an underwater acoustic sensor network in which each node has M_T transmit elements and M_R receive elements (e.g., hydrophones). When a node sends information to another node, its bit stream is split into M_T sub streams and each sub stream is transmitted by one of the M_T transmit elements. All M_T transmit elements transmit sub bit streams simultaneously to the receiver with the same carrier frequency and bandwidth. In a narrowband scenario, the received signal at the receiver can be modeled as

$$Y(t) = \sqrt{P} X(t)H(t) + Z(t), \quad (2)$$

where $Y(t) = [y_1(t) \ y_2(t) \ \cdots \ y_{M_R}(t)]$ is the received signal vector whose component $y_n(t)$, $1 \leq n \leq M_R$, is the received signal at receive element n , $X(t) = [x_1(t) \ x_2(t) \ \cdots \ x_{M_T}(t)]$ is the transmitted signal vector whose component $x_m(t)$, $1 \leq m \leq M_T$, is the transmitted signal from transmit element m , and $Z(t) = [z_1(t) \ z_2(t) \ \cdots \ z_{M_R}(t)]$ is the noise vector whose components are modeled as independent circularly symmetric complex Gaussian random variables with zero mean and unit variance. In (2), $H(t) = \{h_{m,n}(t) : 1 \leq m \leq M_T, 1 \leq n \leq M_R\}$ is the channel matrix whose component $h_{m,n}(t)$ denotes the channel fading coefficient between transmit element m , $1 \leq m \leq M_T$, and receive element n , $1 \leq n \leq M_R$. We assume that the channel matrix is known at the receiver side, but unknown at the transmitter side. The transmitted signal vector $X(t)$ is assumed to satisfy a power constraint $\sum_{m=1}^{M_T} |x_m(t)|^2 = 1$, i.e., the total transmitted power is P no matter how many transmit elements are deployed at the transmit node. Moreover, we assume that the channel is heavily affected by multipath fading (*saturated condition*, see [42]) as it is often the case in shallow water [2].

C. Multiplexing and Diversity Tradeoff

The frequency-dependent attenuation significantly limits the maximum usable frequency and thus the available communication bandwidth [43]. MIMO transmissions is thus an ideal way to increase data rates for underwater acoustic communications, in which independent data streams can be sent out in parallel by multiple transmit elements in the same frequency band. The increased spectral efficiency is termed *multiplexing gain* [44]. At the receiver side, the receiver can demodulate each of the data streams by nulling out the others with a decorrelator [45].

Besides increasing transmission rates, MIMO can also be used to reduce the received signal error probability and hence to improve the communication link reliability. By sending signals that carry the same information through different channels, multiple faded copies of the data information can be obtained at the receiver. Such a redundancy is termed *diversity* [44] and can be quantified in terms of diversity gain d . The average error probability can be reduced in an order of $1/SNR^d$ at high SNR, so the higher the diversity gain, the higher the reliability of the receiver detection.

Therefore, underwater acoustic communications can benefit from MIMO in two aspects: multiplexing gain and diversity gain. Unfortunately, these two gains cannot be optimized independently and there is a tradeoff between them: higher multiplexing gain can be obtained at the price of sacrificing diversity gain, and vice versa. In an RF scenario, for any targeting multiplexing gain r , the maximum diversity gain is [44] $d(r) = (M_T - r)(M_R - r)$, which depends on the numbers of transceiver elements M_T and M_R .

IV. UMIMO-MAC

Let us consider a network of acoustic devices in a multihop environment, and assume that each sensor node is equipped with $M_T = M_R = M$ transceiver elements. For each packet transmission, each device can encode the information bits to be transmitted in k *parallel independent streams*, with $k \in$

$1, 2, \dots, M$. Given the number of independent streams k , and given a family of space-time codes \mathcal{C} , a multiplexing gain $r(k)$ and a diversity gain $d(r)$ are defined according to the multiplexing and diversity tradeoff.

Formally, given the number M of transceiver elements at the transmitter and the receiver, and given a set of space time codes $\underline{\mathcal{C}} = [C_1, C_2, \dots, C_P]$, a set of *transmission modes* $\underline{M} = [M_1, M_2, \dots, M_P]$, with P being the size of the space of transmission modes, are defined between a transmitter and a receiver. Each transmission mode M_i is associated to a *transmission rate* (or simply rate) R_i [bit/s], with $R_1 \leq R_2 \leq \dots \leq R_P$, a multiplexing gain r_i , and a diversity gain d_i , with the transmission rate increasing with the multiplexing gain, and the bit error rate decreasing with increasing diversity gain.

To explore the relevance of the above decision space to protocol design, let us consider a multimedia application a with bandwidth requirement β^a and bit error rate BER^a . We consider a MIMO CDMA environment [46][47], where a node i needs to transmit a packet to a predetermined neighboring node j . To accomplish this, node i needs to: i) limit the near-far effect when it transmits to node j ; and ii) avoid impairing ongoing communications. These two constraints can be formulated as follows:

$$\begin{cases} \frac{P_{i,j}}{TL_{ij}} \geq \Phi^{m_j}(BER_j^a, INR_j) \\ \frac{S_k}{N_k} \geq \Phi^{m_k}(BER_k^a, (INR_k + \frac{P_{i,j}}{TL_{ik}})), \forall k \in \mathcal{K}_i. \end{cases} \quad (3)$$

The first inequality in (3) states that the signal-to-noise ratio (SNR_j) at the receiver j needs to be above the SNR threshold $\Phi^{m_j}(BER_j^a, INR_j)$, i.e., the value that guarantees the bit error rate BER_j^a required by the application a , given the current interference-to-noise ratio INR_j at the receiver, and a choice of transmission mode m_j that determines a multiplexing gain r_j . The SNR at j is expressed as the ratio between the power received at j ($\frac{P_{i,j}}{TL_{ij}}$) and the receiver noise N_j . $P_{i,j}$ [W] represents the power transmitted by i to j when an ideal channel (without multipath, i.e., $A = 0$ dB) is assumed, while TL_{ij} and TL_{ik} are the transmission losses from i to j and from i to $k \in \mathcal{K}_i$, with \mathcal{K}_i being the set of nodes whose ongoing communications may be affected by node i 's transmit power, respectively. Finally, N_j [W] and N_k [W] are the noise power at nodes j and k , respectively. The second inequality in (3) represents the same constraint for all transmitters affected by the communication between i and j . There, S_k [W] represents the received power of the signal being decoded by a receiver $k \in \mathcal{K}_i$. Note that the interference-to-noise ratio at k is expressed as the sum of the interference-to-noise ratio at k plus an additional component caused by i 's transmission to j .

$\Phi^m(\cdot)$ depends on the bit error rate and the interference to noise ratio at the receiver. However, the SNR threshold Φ^m , as expressed by its dependence on m , is also a function of the given choice of transmission mode, i.e., of the multiplexity-diversity tradeoff. Hence, to accommodate the BER requirements of the application, node i has two choices:

- 1) For a fixed transmit power $P_{i,j}$, use a transmission mode m with multiplexing gain r associated to a SNR

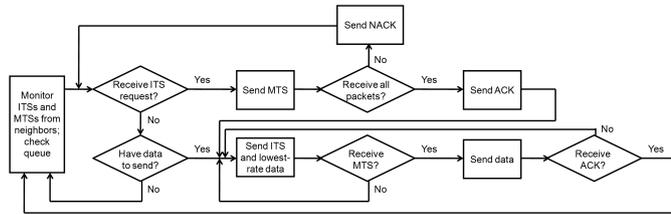


Fig. 1. The flowchart of UMIMO-MAC.

threshold $\Phi^m(\cdot)$ that is low enough to provide the required BER ;

- 2) For a fixed transmission mode m with multiplexing gain r , set its transmit power $P_{i,j}$ to the minimum value that guarantees the required BER .

A. The UMIMO-MAC Protocol

In addition to the objectives previously stated, UMIMO-MAC is designed to reduce the effect of long propagation delays on the channel utilization efficiency, and to efficiently disseminate local information that is needed to make distributed, localized, decisions. We will describe in the following how a suitable transmission mode is selected at the receiver. We refer to Fig. 3, where a transmitter i willing to communicate with a receiver j is depicted. Let us introduce the following:

Definition 1: The upper bound on transmit power P_i^{max} is the maximum transmit power that will not impair ongoing communications for all neighbors of transmitter i . For example, P_i^{max} is the maximum of $P_{i,j}$ in the second inequality in (3).

Definition 2: The lower bound on transmit power $P_{i,j}^{min,m}$ is the minimum transmit power needed to decode a packet at the receiver j with the required BER for a given transmission mode m . As shown in the first inequality in (3), $P_{i,j}^{min,m}$ is the minimum of $P_{i,j}$.

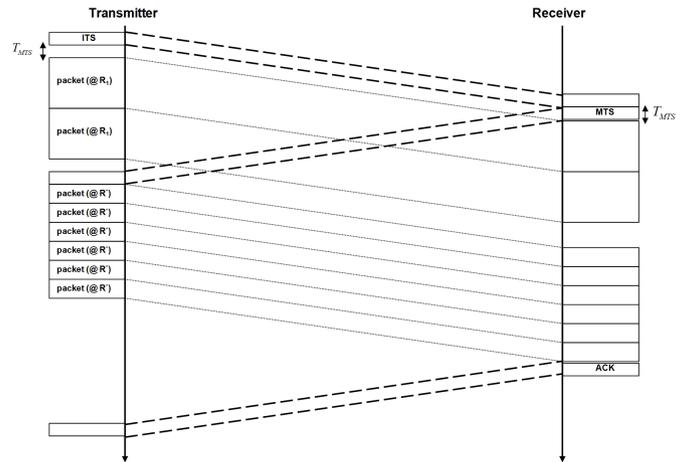
Definition 3: The assigned transmit power $P_{i,j}^*$ is the transmit power assigned to transmitter i after negotiation with receiver j .

Definition 4: The receiver's interference tolerance ΔI_j is the maximum additional interference that will not impair the ongoing communication for receiver j .

Definition 5: The finish receive time t_j is the time at which receiver j will finish receiving packets whose transmission has already been negotiated.

In UMIMO-MAC, each transmitter i is assumed to know the distance $d_{i,j}$ from i to receiver j and the distance $d_{i,k}$ from i to $k \in \mathcal{K}_i$. Each transmitter i is also assumed to be capable of estimating the transmission loss $TL_{i,k}$. Moreover, each receiver j is capable of estimating the multiple access interference (MAI) I_j , noise power N_j , distance $d_{i,j}$, and transmission loss $TL_{i,j}$ between transmitter i and receiver j .

Figure 1 depicts the flowchart of UMIMO-MAC, and Fig. 2 illustrates the basic operations and timing of the UMIMO-MAC protocol. The protocol employs *Intent to Send* (ITS) and *Mode to Send* (MTS) control packets to negotiate and regulate channel access among competing nodes. Note that while this may seem to be analogous to the 802.11-like carrier sense multiple access with collision avoidance protocols (CSMA-CA), the analogy with CSMA-CA is limited to the two-way

Fig. 2. The UMIMO-MAC protocol, where R_1 is the lowest transmission rate and R^* is the assigned transmission rate.

handshake - UMIMO-MAC does not employ carrier sense, and there is no collision avoidance mechanism. In addition, unlike 802.11-like protocols, a single ITS-MTS handshake is used to transmit a block of consecutive packets¹. This is done to improve the utilization efficiency of the underwater channel. ITS and MTS are transmitted using a common spreading code which is known by all nodes.

The ITS contains i) the parameters that will be used by the transmitter to generate the spreading code for the data packet, ii) P_i^{max} , the upper bound on the transmit power, and iii) the total number of packets that will be transmitted back-to-back. Based on this information, the receiver will be able to locally generate the spreading code that the transmitter will use to send data packets. Based on P_i^{max} , the receiver will calculate the appropriate transmit power for the transmitter as will be described in Section IV-E. Besides, by overhearing the ITS, the transmitter's neighbors can become aware of the time when the transmitter will end its transmission.

The MTS contains i) the chosen transmission mode, i.e., the multiplexing and diversity tradeoff, ii) the assigned transmit power $P_{i,j}^*$, iii) the receiver's interference tolerance ΔI_j , and iv) the finish receive time t_j . The chosen transmission mode and the assigned transmit power will be used by the transmitter to generate the signal. However, power and transmission mode are selected at the receiver, since the latter can be responsive to the dynamics of the channel based on local measurements and consequently control loss recovery and rate adaptation. With suitable transmission mode and transmit power obtained by ITS/MTS handshake, the transmitter will not impair ongoing communications, and the receiver will not be impaired by ongoing communications. Therefore, the retransmission probability is reduced, thus avoiding feedback overheads and latency. The receiver's interference tolerance and finish receive time are used by the neighbors of the receiver to determine their own upper bound on transmission power. DATA and ACK are then transmitted using the assigned spreading code. ITS, MTS, and ACK are transmitted using the highest diversity gain, i.e., minimum-rate transmission mode,

¹This is in principle allowed also by 802.11 standards, but in practice very seldom used.

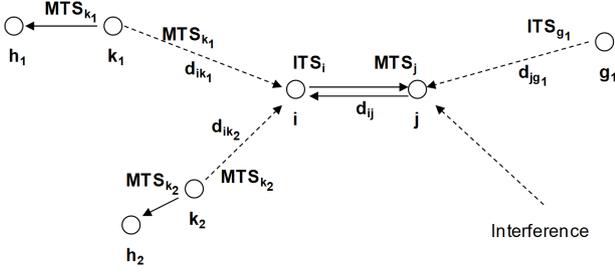


Fig. 3. Message transmissions.

to minimize the packet loss probability. Note that transmission mode and transmit power are selected at the receiver².

A transmitter i continuously monitors the channel for incoming ITSs and MTSs from its neighbors. Based on this, the transmitter infers whether its neighbors are involved in other communications; if this is the case, the time at which each neighbor will finish receiving data. Hence, transmitter i decides when to transmit an ITS according to the information overheard in previous ITS or MTS packets. Three scenarios are possible:

- 1) If no ITS or MTS was received by i from the intended receiver j , i assumes that j is idle, and transmits the ITS immediately;
- 2) If receiver j recently sent an ITS to a node different from i , transmitter i knows that the intended receiver j is currently transmitting data. Transmitter i may or may not know when j will finish transmitting data. If i previously overheard an MTS from the receiver of j 's transmission (for example g_1 in Fig. 3), then i knows j 's finish transmit time. Otherwise, i can estimate the finish transmit time by assuming that j transmits all packets at the lowest-rate transmission mode. Besides, j may retransmit data, and it will transmit another ITS after its finish transmit time. Therefore, i waits the propagation delay time from j to i after j 's finish transmit time, and transmits an ITS immediately if i does not receive any message from its receiver j in order to avoid transmitting while j is still busy and not listening to i . However, the nodes for which receiver j is also their next hop get the same information from j 's ITS. Note that the collision probability of ITSs is low because their propagation delays to the same receiver are different, as will be discussed in Section IV-B. The effect of long propagation delay in underwater reduces the probability of collisions.
- 3) If receiver j previously sent an MTS, transmitter i knows that j is busy receiving data from another node. Hence, i will defer transmission until j 's finish receive time (which i knows since it was contained in the MTS) plus the propagation delay time from j to i . Node i waits the

²Note also that we do not broadcast the rate and remaining number of packets on the common code at the beginning of each data packet to help neighbors better track the remaining transmission time since the remaining transmission time can only be calculated after the receiver chooses the transmission mode, and broadcasting information using the common code at the beginning of each data packet would increase the collision probability of control packets

propagation delay time to prevent j from transmitting to its next hop immediately after it receives data. In UMIMO-MAC, a node transmits the received packet immediately if its next hop is free to reduce the queuing delay and avoid buffer overflows.

During the waiting time, transmitter i can potentially receive ITS and MTS packets from its neighbors and update information on ongoing transmissions accordingly. If another node wants to transmit packets to i , i will defer its transmission schedule and receive these packets first, i.e., nodes accept packets that have already been transmitted to reduce the channel access delay.

The transmitter does not know the actual interference at the receiver side. Thus, the transmitter can only provide information about its upper bound on transmit power. Transmission mode and transmit power are then chosen at the receiver. After transmitting the ITS, instead of just waiting idle for the MTS, which will contain the assigned transmission mode and transmit power, the transmitter starts transmitting packets using the lowest-rate transmission mode data rate. This is done to improve the channel efficiency and thus reduce the effect of the long propagation delays. Immediately after transmitting the ITS, the transmitter waits for T_{MTS} seconds and then transmits packets at the lowest-rate transmission mode with appropriate transmit power, as will be discussed in Section IV-C. T_{MTS} corresponds to the MTS transmission delay plus a turn-around time that is needed by the transceiver electronics to switch between receive and transmit mode, which can be obtained as

$$T_{MTS} = \frac{L_{MTS} \cdot c}{r_c} + T_{elec}, \quad (4)$$

where L_{MTS} [bit] is the MTS size, c [chip/bit] is the spreading code length, r_c [chip/s] is the channel chip rate, and T_{elec} is the turn-around time needed by the transceiver electronics to switch between receive and transmit mode. Note that $\frac{r_c}{c} = R_1$, i.e., the lowest-rate transmission mode. Besides, the number of lowest-rate transmission mode packets n_{LR} can be obtained as

$$n_{LR} = \left\lfloor \frac{2 \cdot \frac{d_{ij}}{\bar{q}} - T_{MTS}}{\frac{L_D \cdot c}{r_c}} \right\rfloor, \quad (5)$$

where \bar{q} is the sound velocity and L_D [bit] is the packet size. In (5), the time interval that can be used to transmit lowest-rate transmission mode packets is $(\frac{2 \cdot d_{ij}}{\bar{q}} - T_{MTS})$, i.e., twice the propagation delay from transmitter i to receiver j minus the time to transmit the MTS, T_{MTS} . The transmission delay of a packet at the lowest data rate is $\frac{L_D \cdot c}{r_c}$. Thus, the number of lowest-rate transmission mode packets is the round towards minus infinity of $(\frac{2 \cdot d_{ij}}{\bar{q}} - T_{MTS}) / \frac{L_D \cdot c}{r_c}$. The receiver will transmit the MTS immediately after receiving the ITS, and the transmitter will transmit packets at the data rate indicated on the MTS at the assigned transmit power immediately after receiving the MTS. After receiving all packets, the receiver will immediately transmit the ACK.

If the transmitter and the receiver are close to each other, i.e., d_{ij} is small, (5) would be zero. The transmitter will start to transmit packets after it receives the MTS when it has 2 or more packets. However, if it has only one packet to send, it will not wait for the MTS. The transmitter will transmit the only packet using the lowest-rate transmission mode, and then

wait for the ACK from the receiver.

If the transmitter receives an ACK requesting retransmissions, it will first wait for a T_{ITS} interval, since the receiver will transmit its ITS immediately after the ACK if its next hop is idle. If no ITS is overheard, the transmitter will transmit its own ITS for retransmission, which will likely not collide with other ITSs because of the effect of long propagation delay. In fact, if multiple nodes intend to use the same next hop, the closest-to-the-next-hop wins the transmission because its lower propagation delay causes its ITS to arrive earlier than all others.

B. Analysis of Control Packet Collisions

Since DATA and ACK packets are transmitted using transmitter-specific spreading codes, collisions only occur when transmitting ITS or MTS packets. However, there is no backoff mechanism in our proposed protocol. In fact, as will be shown in this section, in underwater networks packets retransmitted after a collision are unlikely to collide again. This is caused by the high propagation delay.

A transmitter i retransmits an ITS at the expiration of a timer that is started after transmitting the ITS and set to $T_{MTS} + \frac{2 \cdot d_{ij}}{\bar{q}}$, i.e., the time at which i should received an MTS from receiver j . Conversely, a receiver will not retransmit an MTS when an MTS collides since the corresponding transmitter will retransmit the ITS if the timer expires.

In UMIMO-MAC, the probability that two retransmitted ITSs will collide with one another again is very small. Considering again the scenario depicted in Fig. 3, if d_{jg_1} is larger than d_{ij} , the worst-case scenario occurs when the end of g_1 's ITS collides with the beginning of i 's ITS, and then the beginning of g_1 's retransmitted ITS collides with the end of i 's retransmitted ITS. If the transmission delay of an ITS T_{ITS} is $\frac{L_{ITS} \cdot c}{r_c}$, where L_{ITS} [bit] is the ITS size, the worst-case scenario happens when $T_{MTS} + \frac{2 \cdot d_{jg_1}}{\bar{q}} < T_{ITS} + T_{MTS} + \frac{2 \cdot d_{ij}}{\bar{q}} + T_{ITS}$. Hence, $d_{jg_1} - d_{ij} < T_{ITS} \cdot \bar{q}$. Similarly, if d_{ij} is larger than d_{jg_1} , we have $d_{ij} - d_{jg_1} < T_{ITS} \cdot \bar{q}$. Thus, collisions on retransmitted packets may happen when

$$|d_{ij} - d_{jg_1}| < T_{ITS} \cdot \bar{q}. \quad (6)$$

The expression in (6) shows that a collision on retransmitted packets will never happen if the difference of the distance between the two transmitting nodes and the receiver is larger than $T_{ITS} \cdot \bar{q}$; furthermore, a collision may or may not happen if the distance is about the same. Furthermore, we can state that x additional collisions on retransmissions will never happen when

$$|d_{ij} - d_{jg_1}| > \frac{T_{ITS}}{x} \cdot \bar{q}. \quad (7)$$

For example, if $L_{ITS} = 10$ Bytes, $c = 7$, $r_c = 100$ kcps, and $\bar{q} = 1500$ m/s, the ITSs of the two transmitting nodes with a difference in the distance from the receiver of more than 8.4 m will never collide again after retransmission. The analysis above provides the rationale for not using a backoff mechanism in underwater and in general on propagation media affected by high propagation delay.

C. Upper Bound on the Transmit Power

As discussed above, the transmitter evaluates its upper bound on the transmit power according to local information obtained by overhearing MTSs from its neighbors, as shown in Fig. 3. Thus, the upper bound on the transmit power P_i^{max} is calculated by the transmitter as

$$P_i^{max} = \min[P_i^{max}, \min_{k \in \mathcal{K}_i} (\Delta I_k \cdot TL_{ik} | t_{now} + \frac{d_{ik}}{\bar{q}} < t_k)], \quad (8)$$

where P_i^{max} is the maximum transmit power dictated by hardware constraints, ΔI_k is the interference tolerance of node $k \in \mathcal{K}_i$, t_{now} represents the current time and t_k is the finish receive time of node $k \in \mathcal{K}_i$. After transmitting the ITS, the transmitter will transmit packets at the lowest-rate transmission mode until it receives the MTS from the receiver. The transmit power P_{ij}^{LR} used for lowest-rate packets is

$$P_{ij}^{LR} = P_i^{max} - \Delta P_{th}, \quad (9)$$

where ΔP_{th} is the threshold such that the transmit power will not impair ongoing communications for neighbors, and leave some interference tolerance to them. Since the interference tolerance of the transmitter's neighbors will be very limited if the transmit power is close to the upper bound on transmit power, it is necessary to leave some tolerance for the transmitter's neighbors to overcome additional interference. The transmit power is also defined by (9) for control packets. Since the power in (9) is below the maximum allowable transmit power defined in (8), control packets will not impair ongoing DATA packet transmissions.

D. Lower Bound on the Transmit Power

The receiver evaluates its lower bound on the transmit power according to its perceived interference, as illustrated in Fig. 3. The interference to noise ratio (INR) is $INR_j = \frac{I_j}{N_j}$. Interference coming from nodes one hop away is also considered, but the effect is relatively small since acoustic signals attenuate considerably. Therefore, for a given transmission mode m , $P_{ij}^{min,m}$, which is the minimum power needed to decode packet with the required BER, can be obtained as

$$P_{ij}^{min,m} = \Phi^m(BER_j^a, INR_j) \cdot TL_{ij} \cdot N_j. \quad (10)$$

Figure 4 graphically illustrates steps and variables involved in calculating $P_{ij}^{min,m}$. The plot in Fig. 4 represents the bit error rate (BER) of an underwater acoustic MIMO channel, against varying values of interference-to-noise ratio (on the horizontal axis), for different values of the signal-to-noise ratio and of the transmission mode m (each associated to a multiplexing gain r). Receiver j estimates the INR_j (45 dB in the figure, indicated by a vertical solid line) and has a target BER_j^a of 0.001 (indicated by a horizontal solid line). This defines a set of candidate curves, each of which corresponds to a different allocation of power and choice of a transmission mode, which are able to provide the required target BER for the given interference conditions. If, for each transmission mode, we set lower bound on the transmit power to the value corresponding to the minimum-SNR curve within the set of candidate curves, in the example in the figure we get $\Phi^1 = 32$ dB with $r = 1$, and $\Phi^2 = 35.7$ dB with $r = 2$.

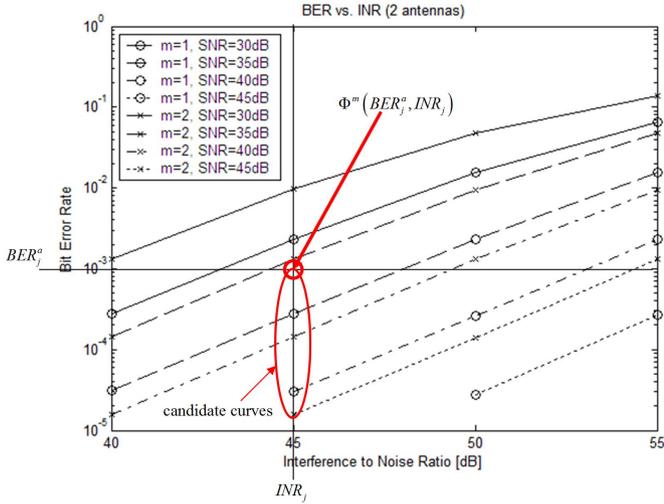


Fig. 4. Lower bound SNR threshold example.

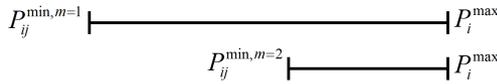


Fig. 5. Upper and lower bounds with different choices of transmission mode.

Therefore, once the desired level of multiplexing gain r is determined (i.e., the transmission mode), the lower bound on the transmit power can be calculated.

E. Joint Selection of Transmission Mode and Transmit Power

Based on the above discussion, transmit mode and power are selected by the receiver. As shown in Fig. 5, while for different choices of transmission mode the upper bound on the transmit power does not vary, the lower bound on the transmit power is different, since resiliency to errors increases with a higher diversity gain and thus allows for a lower transmit power for a given target BER. If the transmit power is chosen to be close to the upper bound, the interference tolerance of transmitter neighbors will be very limited. Conversely, if the transmit power is chosen too close to the lower bound, the interference tolerance of the receiver will be very limited. Thus, it is necessary to avoid transmission modes for which upper bound and lower bound are too close. These considerations lead to the following "Joint Selection of Transmission Mode and Power" strategy as a lexicographic goal programming (LGP) problem [48].

P: Joint Selection of Transmission Mode and Power

Given: $P_i^{max}, P_{ij}^{min, m}, \underline{M}$

Find: m^*, P_{ij}^*

Lex. Maximize: $(r, \Delta P_{ij}^m)$

Subject to: $\Delta P_{ij}^m = P_i^{max} - P_{ij}^{min, m} > \Delta P_{th}$ (11)

The transmit power margin, $\Delta P_{ij}^m = P_i^{max} - P_{ij}^{min, m}$, evaluates how close are the upper and lower bounds for a given transmission mode m . According to different traffic demands, we choose different levels of multiplexing gain r and diversity gain d . For delay-sensitive traffic, e.g., video streams, we first choose the highest allowable r to maximize throughput

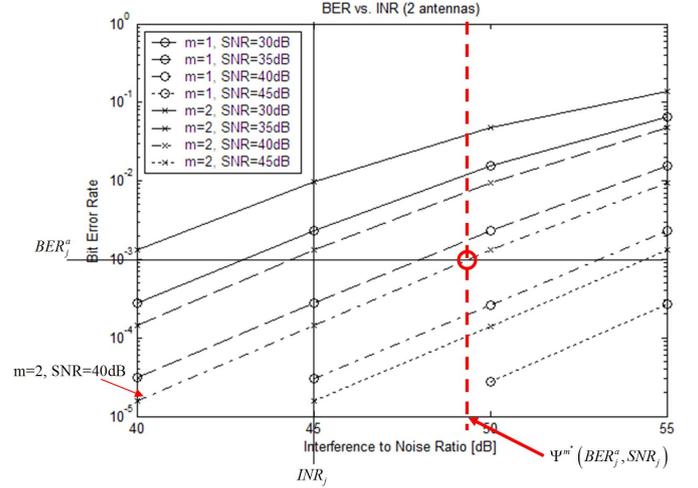


Fig. 6. Interference tolerance example.

and decrease delay. After choosing the multiplexing gain, the transmission mode that yields maximum transmit power margin is selected. Then, the assigned transmit power is calculated as $P_{ij}^* = P_{ij}^{min, m^*} + \Delta P_{tol}$, where $\Delta P_{tol} \leq \Delta P_{th}$ is a suitable margin. Alternatively, for non-real-time data, we can lexicographically maximize $(d, \Delta P_{ij}^m)$ and choose the highest allowable d to minimize power consumption and limit interference to neighboring transmissions³.

F. Interference tolerance and finish receive time

After selecting transmission mode and transmit power, the receiver can calculate the interference tolerance and finish receive time. For a given transmit power, the signal to noise ratio is $SNR_j = \frac{P_{ij}^*}{T L_{ij} \cdot N_j} > \Phi^{m^*}(BER_j^a, INR_j)$. Hence, the receiver has tolerance to overcome additional interference, and the interference tolerance can be obtained as

$$\Delta I_j = (\Psi^{m^*}(BER_j^a, SNR_j) - INR_j) \cdot N_j, \quad (12)$$

where $\Psi(\cdot)$ is the threshold interference-to-noise ratio, which depends on m , BER , and SNR . An example of how to calculate the interference tolerance is shown in Fig. 6. Let us assume that receiver j knows that the selected m^* is 2, SNR_j is 40 dB, BER_j^a is 0.001, and INR_j is 45 dB. Hence, Ψ needs to be 49.2 dB. The interference tolerance of receiver j can then be calculated according to (12). The receiver also informs its neighbors of when it will finish receiving packets. According to Fig. 2, the receiver calculates its finish receive time after receiving the ITS. Hence, it calculates the finish receive time t_j as

$$t_j = t_{now} + T_{MTS} + \frac{2 \cdot d_{ij}}{q} + (n - n_{LR}) \cdot \frac{L_D \cdot c}{r^* \cdot r_c}, \quad (13)$$

where L_D [bit] is the packet size, n is the total number of packets that will be transmitted back-to-back, and n_{LR} is the number of lowest-rate transmission mode packets, as in (5). In (13), the first term t_{now} is the time when receiver j finishes receiving the ITS from transmitter i . The second

³Note that the space of solutions to the above problem is for all practical purposes very limited and can be solved by enumeration - no specialized solver is needed.

term represents the MTS transmission delay, and the third term accounts for the propagation delay from receiver j to transmitter i and from transmitter i to receiver j . The last term represents the transmission delay of the remaining packets. Remember that transmitter i waits for T_{MTS} and then starts transmitting packets using the lowest-rate transmission mode. Hence, some packets are transmitted before transmitter i receives the MTS. Besides, the transmission delay of the assigned transmission mode packet is $\frac{L_D \cdot c}{r^* \cdot r_c}$. Therefore, we can calculate the transmission delay of the remaining packets.

V. PERFORMANCE EVALUATION

We have developed a discrete-event object-oriented packet-level simulator to assess the performance of the proposed cross-layer protocol. MIMO links are simulated by incorporating an acoustic MIMO link module, which we have developed to assess MIMO gains on underwater acoustic links. The physical-layer MIMO link module models the underwater acoustic signal propagation channel with path loss, multipath, and underwater delays. The MIMO link module generates bit error rate curves in terms of input parameters such as the link distance, the numbers of transmit/receive elements, choice of space-time codes, total transmit power, acoustic noise level, Doppler spread and correlation among different channels. For example, Fig. 4 is obtained through our underwater MIMO module and represents a comparison of the bit error rate (BER) of an underwater acoustic link, against varying values of interference-to-noise ratio ($INR_j = \frac{I_j}{N_j}$, on the horizontal axis), for different values of the signal-to-noise ratio (SNR), with a MIMO diversity, MIMO multiplexing, and SISO system, respectively. We considered a MIMO-CDMA environment [46], [47], with fixed length spreading code length 7, and two transmit and receive antennas. The other simulation parameters are the same as described in [6].

We evaluate the performance of UMIMO-MAC in a three-dimensional shallow water environment. In addition, we compare UMIMO-MAC with a simple ALOHA MAC protocol, and against UW-MAC [6]. Note that in [6], UW-MAC had been shown to consistently outperform CSMA/CA (with and without power control), and IEEE 802.11. Finally, to explore the multiplexing-diversity tradeoff, we compare the full UMIMO-MAC protocol with simplified non-adaptive versions of the protocol that transmit at full multiplexing or full diversity. Note that all figures are obtained by averaging over multiple topologies and report 95% confidence intervals. We set the chip rate r_c to 100 kcps, the spreading code length c to 7, the maximum transmission power P^{max} to 10 W, the data packet size to 250 Bytes, ITS, MTS, and ACK size to 10 Bytes. We do not consider or account for a physical-layer preamble in this paper. Still, we believe that the comparison among competing protocols is fair since all would equally need a preamble. In addition, we consider an initial node energy of 1000 J, a maximum number of retransmissions equal to 4, and a queue size of 10 kBytes. All deployed sensors are sources of traffic and are randomly deployed in a 3D shallow water scenario with volume of $500 \times 500 \times 50 \text{ m}^3$. Traffic packets are transmitted to a surface station. In the simulations presented in this Section, two transmission modes are

defined between a transmitter and a receiver, ($d = 2, r = 1$) and ($d = 1, r = 2$), where d is the diversity gain and r is the multiplexing gain. These simple schemes are used in the simulation results reported in Fig. 4. We emphasize, however, that our algorithms can be applied to arbitrary transmission modes, with additional performance gains. The number of hops depends on the distance between source and destination, and the transmission radius varies with transmit power. The propagation delay is more than 10 times compared with the control packet transmission delay in this simulation, and the simulator handles interferences and collisions from all nodes in the network. In Figs. 7 to 9, we evaluate UMIMO-MAC's scalability and resilience to channel collisions by varying the number of sensors. In Figs. 10 to 12, we vary the packet inter-arrival time to measure the effect of traffic.

In the first set of simulations, we set the packet inter-arrival time to 20s to avoid buffer overflows. When the number of sensors increases, the collision probability increases. In Fig. 7, ALOHA is shown to suffer from more collisions and packet retransmissions. UW-MAC⁴ reduces the collision probability, but both ALOHA and UW-MAC are not aware of when the receiver is idle. Hence, it may happen that packets are transmitted while the receiver is busy with another transmission/reception. Thus, the number of packets dropped after exceeding the maximum number of retransmissions is even higher than the number of successfully received packets. In UMIMO-MAC, only ITS and MTS packets can collide, and their size is smaller than a regular data packet size. Conversely, by overhearing ITSs and MTSs from neighbors, UMIMO-MAC decides when to transmit an ITS. This reduces the collision probability, leading to a higher packet delivery ratio. Since UMIMO-MAC Full Multiplexing always selects full multiplexing gain to transmit packets, the transmit power may exceed the upper bound and increase the retransmission probability of ongoing transmission, thus reducing the network throughput.

Figure 8 shows the average delay of successfully received packets. Without up-to-date information from neighbors, ALOHA and UW-MAC do not know if the receiver is idle. This causes retransmission, and increases overall delay for successfully received packets is increased. Moreover, ALOHA and UW-MAC drop a significant amount of packets and their throughput is much lower than UMIMO-MAC. Somewhat surprisingly, the delay of UMIMO-MAC Full Diversity is higher than UMIMO-MAC even though the former always selects full diversity gain. Interestingly, UMIMO-MAC Full Multiplexing shows higher delay than UMIMO-MAC Full Diversity. The higher retransmission probability affects the end-to-end delay because of the high propagation delay in underwater. In Fig. 9, UMIMO-MAC is shown to considerably reduce the energy consumption by selecting suitable transmit power compared with both ALOHA and UW-MAC. The transmitter in UW-MAC knows its neighbors' MAI from periodic broadcast, and selects the transmit power at the transmitter side to reduce the energy consumption. However, it consumes energy on periodic

⁴Since UW-MAC selects the optimal combination of transmit power and code length at the transmitter side relying on local periodic broadcasts of MAI values from active nodes, we set the nodes in UW-MAC broadcast their MAI values every 0.2 s.

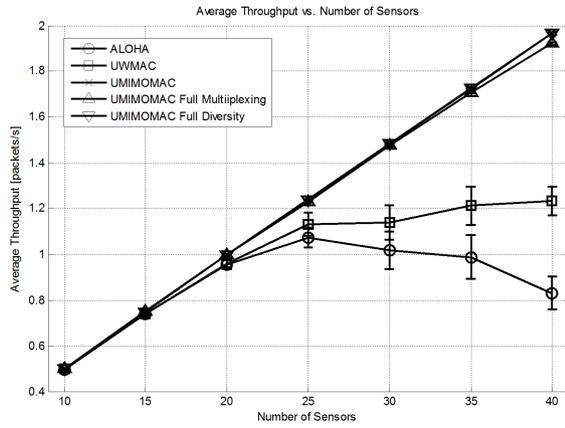


Fig. 7. Average throughput vs. number of sensors.

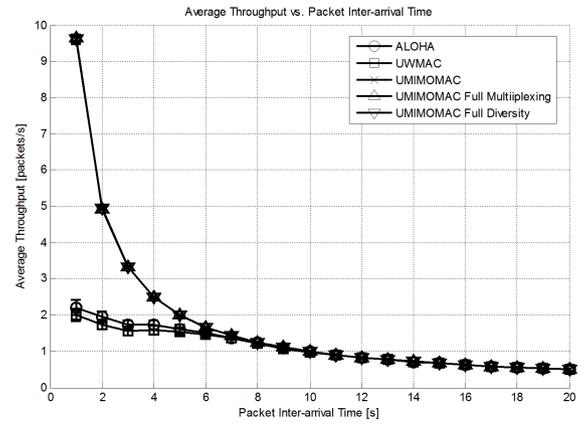


Fig. 10. Average throughput vs. packet inter-arrival time.

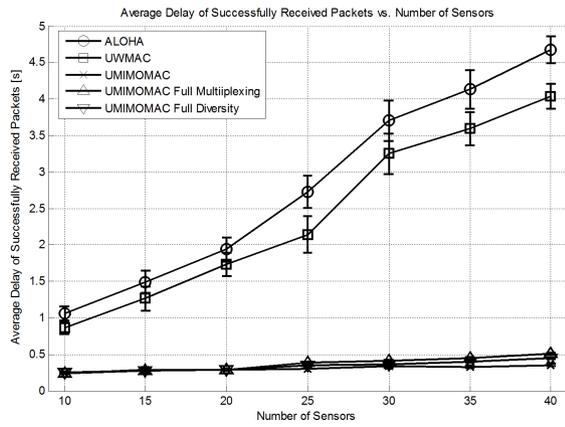


Fig. 8. Average delay of successfully received packets vs. number of sensors.

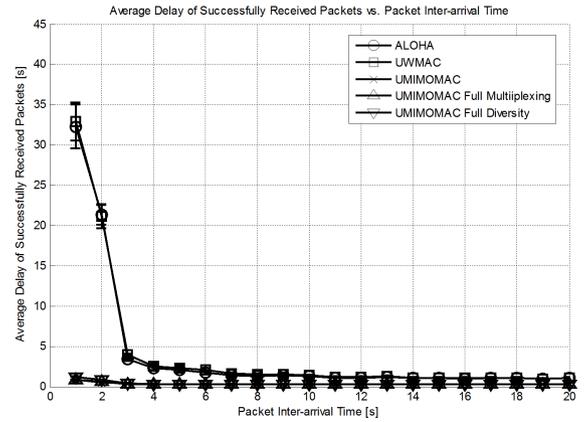


Fig. 11. Average delay of successfully received packets vs. packet inter-arrival time.

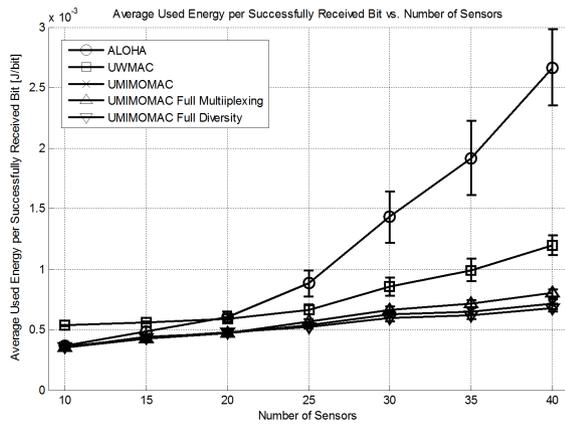


Fig. 9. Average used energy per successfully received bit vs. number of sensors.

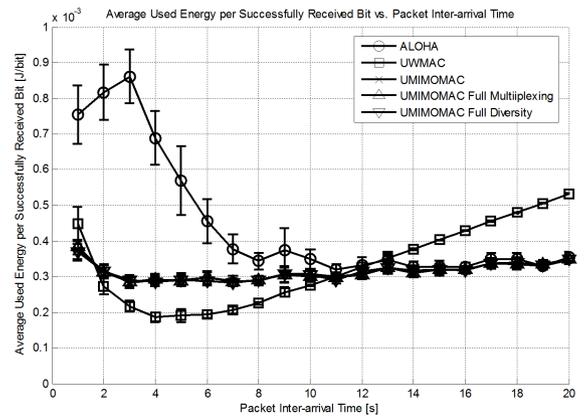


Fig. 12. Average used energy per successfully received bit vs. packet inter-arrival time.

broadcasts, and its performance is lower when the number of sensors is small. Unlike UW-MAC, which depends on periodic broadcasts, UMIMO-MAC selects the transmit power at the receiver side, which provides more accurate information to the transmitter. Moreover, UMIMO-MAC Full Diversity has lower energy consumption than UMIMO-MAC, but its delay is higher.

In Figs. 10 to 12, we set the number of sensors to 10 to avoid a high number of collisions caused by multiple sensors accessing the channel simultaneously. In Fig. 10, the

throughput of UMIMO-MAC is shown to be higher than ALOHA and UW-MAC under heavy traffic. The number of dropped packets caused by buffer overflows or by exceeding the maximum number of retransmissions is still much lower than ALOHA and UW-MAC. Besides, the collision probability is lower in this scenario, and the benefit of UW-MAC in reducing collisions is not significant. UMIMO-MAC transmits a block of consecutive packets, which increases the channel utilization efficiency and consequently the throughput.

Finally, in Fig. 11, we observe that the average delay of

successfully received packets of UMIMO-MAC is lower than ALOHA and UW-MAC under heavy traffic conditions. This is because UMIMO-MAC can transmit multiple packets in a train, thus saving a considerable amount of energy (Fig. 12).

VI. CONCLUSIONS

We proposed, discussed and analyzed a medium access control protocol for underwater acoustic sensor networks with MIMO links. UMIMO-MAC adaptively leverages the tradeoff between multiplexing and diversity gain. Moreover, in a cross-layer fashion, UMIMO-MAC jointly selects optimal transmit power and transmission mode through the cooperation of transmitter and receiver to achieve the desired level of reliability and data rate according to application needs and channel condition. UMIMO-MAC was shown to consistently outperform ALOHA and UW-MAC in terms of network throughput, average delay and energy consumption under several different simulation scenarios.

REFERENCES

- [1] L. Kuo and T. Melodia, "Medium access control for underwater acoustic sensor networks with MIMO links," in *Proc. ACM Intl. Conf. Modeling, Analysis Simulation Wireless Mobile Syst.*, Oct. 2009, pp. 204-211.
- [2] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 257-279, May 2005.
- [3] M. Molins and M. Stojanovic, "Slotted FAMA: a MAC protocol for underwater acoustic networks," in *Proc. MTS/IEEE OCEANS*, Sep. 2006.
- [4] D. Pompili, T. Melodia, and I. F. Akyildiz, "Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks," in *Proc. ACM Intl. Conf. Mobile Comput. Netw.*, Sep. 2006.
- [5] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke, "Data collection, storage, and retrieval with an underwater sensor network," in *Proc. ACM SenSys*, Nov. 2005.
- [6] D. Pompili, T. Melodia, and I. F. Akyildiz, "A CDMA-based medium access control protocol for underwater acoustic sensor networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 4, pp. 1899-1909, Apr. 2009.
- [7] M. O. Khan, A. Syed, W. Ye, J. Heidemann, and J. Wills, "Bringing sensor networks underwater with low-power acoustic communications," in *Proc. ACM Sensys (Demo Session)*, Nov. 2008.
- [8] C. Pelekanakis, M. Stojanovic, and L. Freitag, "High rate acoustic link for underwater video transmission," in *Proc. MTS/IEEE OCEANS*, Sep. 2003, pp. 1001-1007.
- [9] AQUAmodem technology overview. Available: <http://www.aquatecgroup.com/aquamodem.html>
- [10] AquaComm: Underwater wireless modem. Available: http://www.dspcomm.com/products_aquacomm.html
- [11] Link quest. Available: <http://www.link-quest.com/>
- [12] L. Freitag, M. Grund, J. Partan, S. Singh, P. Koski, and K. Ball, "The WHOI micro-modem: an acoustic communications and navigation system for multiple platforms," in *Proc. IEEE Oceans*, Sep. 2005.
- [13] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Computer Netw.*, vol. 51, no. 4, pp. 921-960, Mar. 2007.
- [14] L. Qing-zhong, W. Bing, W. Wen-jin, and G. Xiao-ling, "An efficient underwater video compression algorithm for underwater acoustic channel transmission," in *Proc. IEEE Intl. Conf. Commun. Mobile Comput.*, vol. 2, Jan. 2009, pp. 211-215.
- [15] W. Su and X.-G. Xia, "Signal constellations for quasi-orthogonal space-time block codes with full diversity," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2331-2347, Oct. 2004.
- [16] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction," *IEEE Trans. Inf. Theory*, vol. 44, no. 2, pp. 744-765, 1998.
- [17] D. B. Kilfoyle, J. C. Preisig, and A. B. Baggeroer, "Spatial modulation experiments in underwater acoustic channel," *IEEE J. Oceanic Eng.*, vol. 30, no. 2, pp. 406-415, Apr. 2005.
- [18] B. Li, J. Huang, S. Zhou, K. Ball, M. Stojanovic, L. Freitag, and P. Willett, "MIMO-OFDM for high rate underwater acoustic communications," *IEEE J. Oceanic Eng.*, 2009.
- [19] J. Huang, J.-Z. Huang, C. R. Berger, S. Zhou, and P. Willett, "Iterative sparse channel estimation and decoding for underwater MIMO-OFDM," in *Proc. MITS/IEEE OCEANS Conf.*, Oct. 2009.
- [20] B. Li, S. Zhou, M. Stojanovic, L. Freitag, J. Huang, and P. Willett, "MIMO-OFDM over an underwater acoustic channel," in *OCEANS 2007*, Oct. 2007.
- [21] M. Stojanovic, "MIMO OFDM over underwater acoustic channels," in *Proc. 43rd Asilomar Conf. Signals, Syst. Comput.*, Nov. 2009.
- [22] A. Radošević, D. Fertoni, T. Duman, J. Proakis, and M. Stojanovic, "Capacity of MIMO systems in shallow water acoustic channels," in *Proc. 43rd Asilomar Conf. Signals, Syst. Comput.*, Nov. 2009.
- [23] A. Roy, T. Duman, L. Ghazkhani, V. McDonald, J. G. Proakis, and J. Zeidler, "Enhanced underwater acoustic communication performance using space-time coding and processing," in *Proc. Oceans Conf.*, vol. 1, Nov. 2004, pp. 26-33.
- [24] H. C. Song, W. S. Hodgkiss, and W. A. Kuperman, "MIMO time reversal communications," in *Proc. 2nd Workshop Underwater Netw.*, Sep. 2007, pp. 5-10.
- [25] G. Palou and M. Stojanovic, "Underwater acoustic MIMO OFDM: an experimental analysis," in *Proc. IEEE Oceans Conf.*, Oct. 2009.
- [26] B. Li, J. Huang, S. Zhou, K. Ball, M. Stojanovic, L. Freitag, and P. Willett, "Further results on high-rate MIMO-OFDM underwater acoustic communications," in *OCEANS 2008*, Sep. 2008.
- [27] Y. H. Kwang, B. Sharif, A. Adams, and O. Hinton, "Implementation of multiuser detection strategies for coherent underwater acoustic communication," *IEEE J. Oceanic Eng.*, vol. 27, no. 1, pp. 17-27, Jan. 2002.
- [28] M. Stojanovic, "Recent advances in high-speed underwater acoustic communications," *IEEE J. Oceanic Eng.*, vol. 21, pp. 125-136, Apr. 1996.
- [29] P. Xie and J. Cui, "Exploring random access and handshaking techniques in large-scale underwater wireless acoustic sensor networks," in *Proc. IEEE/MTS OCEANS*, Sep. 2006.
- [30] —, "R-MAC: an energy-efficient MAC protocol for underwater sensor networks," in *Proc. International Conf. Wireless Algorithms, Syst., Appl.*, Aug. 2007.
- [31] K. B. Kredon and P. Mohapatra, "A hybrid medium access control protocol for underwater wireless networks," in *Proc. ACM Intl. Workshop Underwater Netw.*, Sep. 2007.
- [32] A. F. Harris and M. Zorzi, "On the design of energy-efficient routing protocols in underwater networks," in *Proc. IEEE Intl. Conf. Sensor Ad-hoc Commun. Netw.*, June 2007.
- [33] M. Stojanovic, "On the relationship between capacity and distance in an underwater acoustic channel," in *Proc. ACM Intl. Workshop Underwater Netw.*, Sep. 2006.
- [34] W. Zhang and U. Mitra, "A delay-reliability analysis for multihop underwater acoustic communication," in *Proc. ACM Intl. Workshop Underwater Netw.*, Sep. 2007.
- [35] C. Carbonelli, S.-H. Chen, and U. Mitra, "Error propagation analysis for underwater cooperative multihop communications," *Elsevier J. Ad Hoc Netw.*, vol. 7, no. 4, pp. 759-769, June 2009.
- [36] Z. Zhou and J.-H. Cui, "Energy efficient multi-path communication for time-critical applications in underwater sensor networks," in *Proc. ACM Intl. Symp. Mobile Ad Hoc Netw. Comput.*, 2008, pp. 221-230.
- [37] K. Sundaresan, R. Sivakumar, M. A. Ingram, and T.-Y. Chang, "Medium access control in ad hoc networks with MIMO links: optimization considerations and algorithms," *IEEE Trans. Mobile Comput.*, vol. 3, no. 4, pp. 350-365, Oct.-Dec. 2004.
- [38] J.-S. Park and M. Gerla, "MIMOMAN: a MIMO MAC protocol for ad hoc networks," *Lecture Notes Comput. Science*, vol. 3738, pp. 207-220, 2005.
- [39] A. Muqattash and M. Krunz, "CDMA-based MAC protocol for wireless ad hoc networks," in *Proc. ACM Intl. Symp. Mobile Ad Hoc Netw. Comput.*, June 2003, pp. 153-164.
- [40] R. J. Urick, *Principles of Underwater Sound*. McGraw-Hill, 1983.
- [41] F. Fisher and V. Simmons, "Sound absorption in sea water," *J. Acoustical Society America*, vol. 62, no. 3, pp. 558-564, Sep. 1977.
- [42] J. G. Proakis, *Digital Communications*. McGraw Hill, 2001.
- [43] R. Coates, *Underwater Acoustic Systems*. John Wiley & Sons Inc., 1989.
- [44] L. Zheng and D. N. C. Tse, "Diversity and multiplexing: a fundamental tradeoff in multiple-antenna channels," *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1073-1096, May 2003.
- [45] G. J. Foschini, G. Golden, R. Valenzuela, and P. Wolniansky, "Simplified processing for high spectral efficiency wireless communication employing multi-element arrays," *IEEE J. Sel. Areas Commun.*, vol. 17, pp. 1841-1852, Nov. 1999.
- [46] R. L. Choi, R. D. Murch, and K. B. Letaief, "MIMO CDMA antenna system for SINR enhancement," *IEEE Trans. Wireless Commun.*, vol. 2, no. 2, pp. 240-249, Mar. 2003.

- [47] T. S. Dharma, A. S. Madhukumar, and A. B. Premkumar, "MIMO block spread CDMA systems for broadband wireless communications," *IEEE Trans. Wireless Commun.*, vol. 7, no. 6, pp. 1987-1992, June 2008.
- [48] K. Miettinen, *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.



Li-Chung Kuo received his B.S. degree in Electrical Engineering from National Taiwan University, Taipei, Taiwan in 1999. He received his M.S. degree in Communication Engineering from National Chiao Tung University, Hsinchu, Taiwan in 2005. He received his M.S. degree in Electrical Engineering from The State University of New York at Buffalo, Buffalo, NY in 2008. He is currently a Ph.D. student in Wireless Networks and Embedded Systems Laboratory, Electrical Engineering, The State University of New York at Buffalo. He is pursuing his Ph.D.

degree under the supervision of Dr. Tommaso Melodia.



Tommaso Melodia (M'2007) (tmelodia@eng.buffalo.edu) is an Assistant Professor with the Department of Electrical Engineering at the University at Buffalo, The State University of New York (SUNY). He received his Ph.D. in Electrical and Computer Engineering from the Georgia Institute of Technology in 2007. He had previously received his "Laurea" (integrated B.S. and M.S.) and Doctorate degrees in Telecommunications Engineering from the University of Rome "La Sapienza," Rome, Italy, in 2001 and 2006,

respectively. He coauthored a paper that was recognized as the Fast Breaking Paper in the field of Computer Science for February 2009 by Thomson ISI Essential Science Indicators, and a paper that received an Elsevier Top Cited Paper Award. He is an Associate Editor for the *Computer Networks* (Elsevier), *IEEE COMMUNICATIONS SURVEYS AND TUTORIALS*, and for the *Journal of Sensors* (Hindawi). He serves in the technical program committees of several leading conferences in wireless communications and networking, including IEEE Infocom, ACM Mobicom, and ACM Mobihoc. He was the technical co-chair of the Ad Hoc and Sensor Networks Symposium for IEEE ICC 2009. His current research interests are in underwater acoustic networking, cognitive and cooperative networking, and multimedia sensor networks.