M-FAMA: A Multi-session MAC Protocol for Reliable Underwater Acoustic Streams

Seongwon Han, Youngtae Noh†, Uichin Lee‡, Mario Gerla
Dept. of Computer Science, UCLA †Cisco Systems Inc. ‡KSE, KAIST, Korea
{swhan, gerla}@cs.ucla.edu, †ynoh@cisco.com, ‡uclee@kaist.edu

***Abstract*—Mobile underwater networking is a developing technology for monitoring and exploring the Earth's oceans. For effective underwater exploration, multimedia communications such as sonar images and low resolution videos are becoming increasingly important. Unlike terrestrial RF communication, underwater networks rely on acoustic waves as a means of communication. Unfortunately, acoustic waves incur long propagation delays that typically lead to low throughput especially in protocols that require receiver feedback such as multimedia stream delivery. On the positive side, the long propagation delay permits multiple packets to be "pipelined" concurrently in the underwater channel, improving the overall throughput and enabling applications that require sustained bandwidth. To enable session multiplexing and pipelining, we propose the Multi-session FAMA (M-FAMA) algorithm. M-FAMA leverages passively-acquired local information (i.e., neighboring nodes' propagation delay maps and expected transmission schedules) to launch multiple simultaneous sessions. M-FAMA's greedy behavior is controlled by a Bandwidth Balancing algorithm that guarantees max-min fairness across multiple contending sources. Extensive simulation results show that M-FAMA significantly outperforms existing MAC protocols in representative streaming applications.**

***Index Terms*—Underwater, AUV, SEA Swarm, Medium Access Control, Concurrent Transmission, CSMA**

I. INTRODUCTION

Although oceans cover two-thirds of the Earth's surface, human exploration and understanding of these frontiers has historically been limited by technical barriers. Recent work suggests that Underwater Acoustic Sensor Networks (UW-ASNs) are effective tools for exploring and observing the ocean [1], [2], [3]. An example is the SEA Swarm (Sensor Equipped Aquatic Swarm) architecture, where a large number of sensors are deployed as a group that moves with the water current [4], [5] (see Fig. 1). A swarm of sensor nodes is escorted by surface relay buoys, which are equipped with acoustic, RF, and satellite interfaces. Each sensor monitors local underwater activities and acoustically reports critical multimedia data to any one of the surface stations over multiple hops if necessary. The data are then relayed via radio channels from the buoys to a central monitoring station.

Swarm mobility presents new technical hurdles—especially in the context of an acoustic communications channel [6]. Despite technological advances in acoustic communications, several challenges remain, including: limited bandwidth, long propagation delay ($1.5km/s$: five orders of magnitude slower than radio frequency) [7], relatively high transmission energy cost (with typical reception vs. transmission energy ratio of 1:125 [8]). Many underwater MAC protocols have recently been proposed to address such limitations. Most of these protocols are based on Carrier Sense Multiple Access (CSMA) or Code Division Multiple Access (CDMA). Only a few use Time Division Multiple Access (TDMA) or Frequency Division Multiple Access (FDMA). TDMA necessitates a network-wide time consensus, which results in a large number of control packet exchanges and requires a lengthy synchronization process. This implies that all nodes must remain synchronized, regardless of node failures or node movements, in order to maintain reliable transmission schedules. More importantly, TDMA-based methods are not suitable for resource-constrained underwater mobile sensor networks, because nodes must periodically perform expensive scheduling operations [9], [10], [11], [12], [13]. Likewise, FDMA is an inherently inefficient protocol for UW-ASNs; only a subset of the available frequency/bandwidth can be used due to the prevalent fading in underwater environments [14], [15], [16]. Since CSMA-based protocols easily support the required network dynamics (i.e., node mobility, failure, joining, and leaving), interest in CSMA-based protocols has increased recently over CDMA [13]. Long propagation delays cause significant performance degradation in protocols using a ready-to-send/clear-to-send (RTS/CTS) mechanism. Newer CSMA-based protocols attempt to address this challenge, by enabling channel reuse through concurrent transmission sessions [17], [18], [19], [20], [21]. Here, channel reuse is typically categorized as either *temporal* or *spatial* reuse; i.e., temporal reuse occurs when multiple outstanding packets can be scheduled without collisions (either from single sender or multiple senders), and spatial reuse occurs when multiple neighboring (exposed) terminals transmit at the same time. However, to the best of our knowledge, none of the existing protocols fully exploit the channel reuse properties underwater. The channel reuse in recent protocols such as RIPT [20] and DOTS [21] is limited to the receiver side. While each

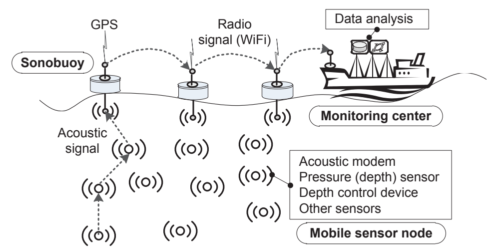


Fig. 1. SEA-Swarm Architecture

This work is supported in part by the US National Science Foundation under Grant No. 1205757 and the National Research Foundation of Korea Grant funded by the Korean Government (MEST) (NRF-2012R1A1A1008858).

receiver supports multiple sessions from different neighboring senders, there is no support for a sender to initiate multiple sessions to the other nodes (also known as pipelining). APCAP attempts to enable multiple sessions at the sender side by transmitting packets out-of-order [17], but it does not detail scheduling strategies for out-of-order packet delivery. In fact, full packet pipelining support is challenging in that explicit flow/congestion control must be implemented at the MAC layer in a distributed fashion—a simple back-off scheme is not sufficient. So far none of the existing protocols considered this issue.

In this paper, we propose a new underwater MAC protocol called Multi-session Floor Acquisition Multiple Access (M-FAMA) that permits senders to initiate multiple concurrent *sessions*[1] to other receivers. Sources avoid collisions by calculating their neighbors' transmission schedules and propagation delays from passively overheard message transmissions. By adding a small *guard time* to calculated transmission schedules, M-FAMA protects against collisions arising from node mobility (as depicted in Fig. 1). M-FAMA was inspired by the DOTS protocol and makes three important contributions beyond DOTS: 1) it allows multiple outgoing sessions from each source and multiple (pipelined) packets on each session; 2) it applies a localized distributed algorithm (called bandwidth balancing) to maintain max-min fairness between sessions within the same collision domain; 3) it shows significant gains with respect to DOTS when applied to a representative underwater monitoring and surveillance scenario. The remainder of the paper defines the M-FAMA protocol and reports extensive simulation experiments comparing M-FAMA with existing underwater MAC protocols.

II. M-FAMA: Motivations and Basic Principles

An RTS/CTS exchange can be used to reduce the chance of collision interference due to the hidden terminal problem [22]. However, this solution does not prevent collisions entirely—especially in a high-latency environment like the Underwater Acoustic scenarios. Fullmer found that imposing wait times on RTS/CTS transmissions can reduce collisions in cases of channel contention [23]. Moreover, he identified the following two conditions for collision-free transmission when using an RTS/CTS mechanism:

- *RTS wait time*: The time from RTS transmission to receiver issue of CTS response – should be greater than the maximum propagation delay (the time for a transmitted frame to reach its maximum transmission range)
- *CTS wait time*: The time a sender waits to receive a CTS reply after transmitting its RTS – should be greater than the RTS transmission time plus double the maximum propagation delay plus the hardware transmit-to-receive transition time.

Based on this work, Molins *et al.* proposed Slotted-FAMA [24] for underwater networks where node communications are slot-synchronized, and any packet exchanges (including RTS/CTS/ACK control packets) can only happen at the beginning of a slot. While the protocol experiences fewer collisions,

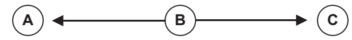


Fig. 2. Topology

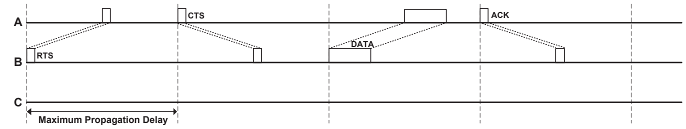


Fig. 3. Slotted FAMA

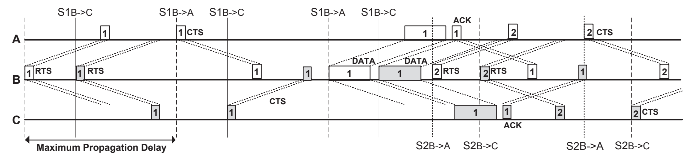


Fig. 4. Multi-session FAMA

guard times within slots and fixed slot size exacerbate the delays between communications.

Propagation delays increase collisions and reduce the channel throughput when using an RTS/CTS mechanism. Fortunately, large propagation delay also creates a new opportunity to achieve higher throughput by reusing the channel with interleaved sessions. Note that collisions only occur at the receiver side, not the sender side [25]. A collision occurs when two or more signals arrive simultaneously at a receiver, which is unable to decode the overlapped signals. In light of this definition, we relax the constraint imposed by most MAC protocols that the sender be protected from interference. Opportunistic concurrent transmissions, when the transmitter is "exposed", can improve the throughput significantly. Fig. 2 shows a network topology useful for illustrating the benefits of channel reuse in M-FAMA. In this case, node A and node C are hidden from each other, but node B is within the transmission range of A and C. As depicted in Fig. 3, Slotted FAMA restricts channel access to only one sender-receiver pair (e.g., node B and A) during the slot time. Slot time is determined by the maximum propagation delay, which is 0.5 second for the $750m$ communication range in an UW-ASN. With no channel reuse and a $16kbps$ data rate acoustic connection, sending a $128byte$ (or $1024bits$) data packet during a RTS-CTS-DATA-ACK (i.e., session) represents a worst-case channel utilization of 3.2%.

Multi-session FAMA leverages the RTS/CTS exchange for learning propagation times to and between neighbors. With loose clock synchronization among terminals and a known transmission and propagation time for each control and DATA packet (assumed of fixed size), each terminal can calculate its neighbors' transmission and reception schedules by promiscuously overhearing the neighbor's transmissions. Using the knowledge of neighbors' schedules, a node can schedule collision-free transmissions of its own.

Fig. 4 shows how M-FAMA leverages this knowledge to actively initiate four sessions for two receivers; in the same amount of time, S-FAMA (Fig. 3) is only able to transfer one RTS-CTS-DATA-ACK sequence to a single receiver. In the case depicted in Fig. 4, node B first transmits an RTS destined for node A. While the RTS packet is still propagating, node B then transmits another RTS destined for node C. When node A receives its RTS, it waits until time (packet transmission time + maximum propagation delay) and then replies with a CTS. Meanwhile, node C has also received its RTS, and replies

[1]The term *session* conventionally refers to opening, closing, and managing a communications dialogue between end-user application processes. In this paper, we use it to describe the time during which an RTS-CTS-DATA-ACK sequence is exchanged between a sender and its intended receiver.

by transmitting CTS after waiting the appropriate amount of time. Node B receives CTS messages from nodes A and C sequentially, and then sends the necessary DATA messages consecutively. Note that node B actively initiates second sessions for both nodes A and C by sending another RTS to the destination before receiving an ACK from nodes A or C for the previous session's DATA transmissions. The reader will notice that the number of simultaneous sessions from the central node to the peripheral nodes can be increased as the ratio $\{propagation\ time/transmission\ time\}$ increases.

To motivate the use of M-FAMA in more general topologies, consider the application depicted in Fig. 1. Several underwater sensors are crawling the bottom of the sea, in part carried by the underwater currents, mapping the habitat, say, with video cameras. The video is low resolution, possibly a sequence of still images. It is stored at the underwater source and delivered as a multimedia file to sonobuoys; from these it is forwarded to the support ship via radio. Swarm nodes, sonobuoys and ship can move with currents. The continuous motion prevents the use of tethers (i.e., fibers and/or cables) that connect sensors to a support ship. Instead, untethered node-to-node and node-to-buoy communications are used. The dynamic underwater sink tree that connects various sea bottom crawlers to one sonobuoy must multiplex several streams from several different sources. More precisely, at each intermediate repeater, multiple streams (from different sources) must be forwarded upwards. This causes an intermediate node in one tree to be exposed to interference from other trees. We will show that M-FAMA spatial multiplexing feature can efficiently handle these problems. Moreover, M-FAMA multi-session pipelining will be very effective with links experiencing large propagation/transmission delay ratios. The experiment will demonstrate the throughput gains and video improvements yielded by M-FAMA over existing protocols.

The following section provides a detailed description of how M-FAMA performs scheduling and maintenance, and presents two variants (conservative and aggressive) of the M-FAMA protocol.

III. Multi-session FAMA design

As previously mentioned, M-FAMA nodes monitor MAC-level state information to avoid collisions. This information includes transmission and reception schedules affecting one-hop neighbors, and a delay map indicating propagation delays between the node and its one-hop neighbors, and between one- and two-hop neighbors. To allow state information to be assembled entirely through passive channel observation, we enhance the MAC frame headers with supplementary data (discussed in III-A). Based on the type of MAC frame overheard (e.g., RTS, CTS, DATA, ACK) and the delay map information, a node is able to infer when its one-hop neighbors will be receiving transmissions. This information allows the node to avoid opening a session that would collide with a neighbor's message reception. Similarly, combining the node's knowledge of its current sessions with the delay map information allows it to calculate the times when it will be receiving packets. Thus, the node can avoid opening sessions that would conflict with its own receptions. To avoid scheduling errors caused by node mobility or clock skew, we incorporate a motion-dependent guard time, which will be further discussed in Section III-A.

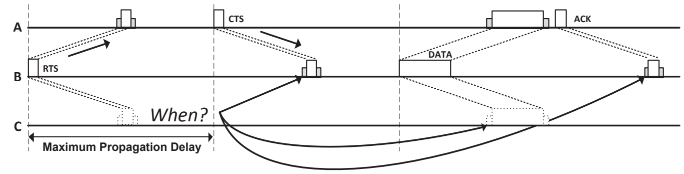


Fig. 5. Node C's delay map update after overhearing node B's RTS

Whenever a node has a frame to send, it compares the transmission against neighboring and local reception times to detect potential collisions (Section III-B). If no conflicts are detected, the node begins its transmission; otherwise, it backs off the communication. Since acoustic fading/scattering may interfere with overheard transmissions, collisions cannot be entirely eliminated through this mechanism. M-FAMA provides an optimized recovery scheme (Section III-D). Further, given that fairness is not guaranteed (e.g., some sources with high data rate could capture an unfair fraction of the channel), we enforce fairness with a "bandwidth balancing" policy across all sources (Section III-E).

A. Delay Map Management and Guard Time

For a node to build the *delay map* through passive listening to the channel, each frame header must contain the following information:

- *source address*: the sender of the observed MAC frame
- *destination address*: the intended destination for the observed MAC frame
- *transmission timestamp*: the time at which the observed MAC frame was sent
- *src-dest delay*: the estimated propagation delay between the source and the destination

By inspecting overheard frames from neighboring transmissions, an M-FAMA terminal is thus able to construct the delay map of the propagation delays with its one-hop neighbors, and between its one- and two-hop neighbors. Combining the delay information with type of the message overheard, the node can predict future transmissions and receptions in the channel. Whenever the node's MAC layer seeks to transmit a frame, it calculates the transmission and reception times for all the messages (RTS, CTS, DATA, and ACK) required for successful transmission to the destination terminal. If any of these communications would collide with any neighboring or local receptions, the node refrains from transmitting, backing off its communications until no further collisions can occur.

Noh *et al.* previously demonstrated that time synchronization can be efficiently achieved and maintained in underwater acoustic networks [21]. With clock synchronization across sensor nodes, the value of the timestamp provides timing information for each frame; each node can calculate the propagation delay to a neighbor by subtracting the transmission timestamp of the MAC frame from the reception time of the MAC frame. Using this information, the overhearing node can deduce the expected times when it will be overhearing future communications for the session, and when its neighbors will be receiving transmissions of their own. This process is depicted in Fig. 5, where node C overhears an RTS message from node B for an intended communication with node A. Node C calculates B's expected CTS reception time as the RTS timestamp + maximum propagation delay + estimated

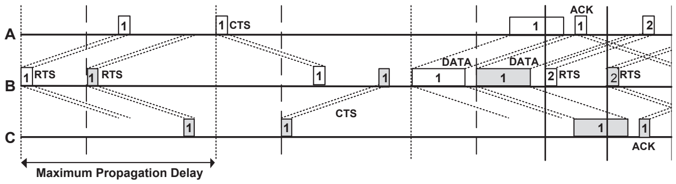


Fig. 6. M-FAMA *conservative*

src-dest delay. Node C can also estimate when it will overhear B's data transmission, which is node B's RTS timestamp + 2 * maximum propagation time + propagation delay between node B and C. Finally, node C determines B's expected ACK reception time as node B's RTS timestamp + 2 * maximum propagation time + hardware receive-to-transmit transition time + expected src-dest delay. This information is incorporated into C's delay map and neighboring transmission schedules, allowing C to avoid transmissions that would lead to collisions at its neighbors, or where C would need to receive data at the same time a neighboring message would be overheard.

To cope with node mobility caused by the ocean currents, M-FAMA introduces a guard time in its calculations. The basic idea is to estimate the displacement of the mobile node between two subsequent control packet advertisements and use a corrective term, called guard time, to account for such error. Assuming that each node will announce itself after an interval proportional to the max acoustic signal RTT between any two nodes, the displacement will be equal to the distance covered in the above interval. Thus, each node calculates this guard time as $2 * (average\ movement\ distance/speed\ of\ sound\ in\ water)$. Note that the multiplier (2) is used to account for the case where the sender and the receiver are moving in opposite directions. This guard time is then added to the end time for the frame reception in the delay map, providing a protection against collisions resulting from node mobility.

B. Delay-map Assisted Packet Scheduling

Whenever a node initiates a communication or overhears RTS or CTS transmission from its neighbors, the node calculates timing information for all future transmissions and receptions in the sequence. The node creates entries in its delay map for each of the calculated times. Before sending an RTS for a new session, the node inspects its delay map to verify that all the components associated with this session (i.e., its own RTS and DATA; CTS and ACK from receiver) will not interfere with neighbors' activities. Moreover it must verify that impending RTS and DATA packets from other nodes will not interfere with its own receiver reception. If these conditions are met, the node proceeds with the communication.

C. Enabling Multiple Sessions

The M-FAMA MAC layer maintains a buffer where it queues packets received from the Network layer. This allows us to avoid Head-of-Line blocking (HOL) without violating protocol layering by directly accessing the network-layer queue to find unblocked prospective sessions with other destinations. From the sender's perspective, this MAC-layer queuing permits a new session to be created whenever a packet received from the Network layer is allowed to be transmitted. New sessions can be established sequentially to the same receiver—this is the feature that we generally call pipelining. They are all called sessions in this context. When the sender session is created, the node compares transmission and reception schedules for the new session against the existing communication schedules. If a collision is anticipated, the state's timing information is reset, and the node will reattempt the transmission session after a backoff period. Otherwise, the node will send the RTS packet for this session and wait to receive a CTS packet. When the CTS packet arrives, the node sends a DATA packet and waits for an ACK packet to arrive. The ACK packet signals completion of the session, meaning that the session has terminated.

For the receiver, a new session is created whenever the node receives an RTS. When the receiver session is created, the node compares transmission and reception schedules for the new session against the existing communications schedules. If a collision is anticipated, the session is closed and the sender will need to send another RTS to attempt the session again. Else, the receiver returns the CTS packet for this session and waits to receive the DATA packet from the sender. When the DATA packet arrives, the node extracts the Network-layer packet and sends it up to the Network layer. The receiver sends an ACK packet and terminates the session.

There are two M-FAMA variants, $Conservative$ and $Aggressive$. They differ in terms of when senders are permitted to open new sessions. Fig. 6 and Fig. 7 show how multiple sessions are maintained in these two M-FAMA variants. The M-FAMA $Conservative$ mode penalizes pipelining in favor of spatial multiplexing. Namely, the node is not allowed to start the next session for the same destination until it transmits the first session's DATA packet. However, it can freely open new sessions with different destinations, taking advantage of spatial reuse. For example, in Fig. 6, the sending node B cannot open a new session for the node A, but it can open a new session with a different destination, e.g. node C. Since the second session for the same destination is allowed only after transmission of the DATA packet for the first session, the sending node will never have more than two active sessions per destination. This feature is meant to improve fairness, by preventing one receiver from monopolizing the sender, while the other potential receivers are starved. The conservative mode is well suited to networks with rich fan out and relatively short propagation delays.

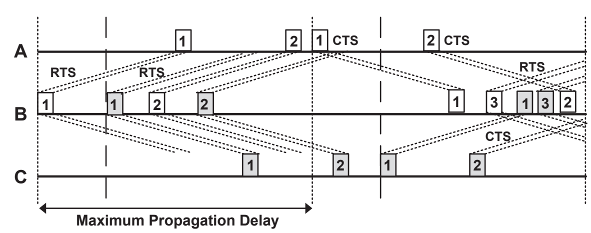


Fig. 7. M-FAMA *aggressive*

In contrast, M-FAMA $Aggressive$ permits to pipeline more sessions to the same destination. The next transmission is scheduled after the transmission of the previous RTS or reception of the CTS to this destination. In addition, new sessions are allowed immediately after the transmission of a previous session's DATA packet to this destination. This allows a sender to open as many sessions per destination as the propagation delay permits. We see this in Fig. 7, where B opens a connection with A and C, and then opens a second connection with both nodes without waiting for a CTS reply. M-FAMA $Aggressive$ is designed to provide higher throughput in cases of low channel contention and high propagation delays (where pipelining is essential).

D. Backoff, Recovery, and Maintenance

Since each node checks against its delay map before starting a new session, new sessions cannot collide with existing sessions (provided that all RTS/CTS messages were overheard successfully). Control packet collisions can occur, for instance when RTS messages from different senders arrive at the same node. However, as the RTS packet size is very small, the probability of such a collision is very small. M-FAMA uses a Binary Exponential Backoff (BEB) algorithm to recover from collisions. Given the long propagation delays and high error rates in our network, exponential growth in the Contention Window (CW) from a pure BEB scheme leads to an overly large window size. Therefore, we use a modified BEB scheme: each node starts a timer after its RTS transmission, and counts the overheard RTS packets from other nodes until the timer expires before receiving its own CTS. This count provides a heuristic estimate of the number of other nodes contending for the channel. We denote this number as the Observed Contender Count (OCC), similar to the counter in the backoff scheme proposed in T-Lohi [26]. Therefore, our backoff algorithm can be expressed as follows:

$$\begin{cases} CW = min(2 \times CW,\ CW_{min} 2^{OCC}) & \leftarrow \text{ upon collision} \\ CW = CW_{min} & \leftarrow \text{ upon success} \end{cases} \tag{1}$$

The receiver upon receiving an RTS, will wait until a time equal to (RTS transmission timestamp + maximum propagation time) before sending a CTS response with the earliest packet creation time. This approach basically eliminates all spatial unfairness.

Each M-FAMA node uses a refresh and expiration mechanism to account for backed-off or canceled neighbor transmissions, and stale delay information for its one-hop neighbors. Whenever a new transmission is overheard, the node searches its state information for entries with matching source and destination fields. In the case of duplicate entries, the node keeps the entry with the latest timestamp. An expiration timer is set for each entry added to the delay map and scheduled transmissions. When the timer expires, the item is removed, to limit the delay map and transmission schedule overhead.

E. Bandwidth balancing

M-FAMA relies on the delay map to minimize collisions. At the same time, it greedily maximizes throughput within the given delay map constraints (especially if the aggressive multiplexing approach is used). However, fairness is not guaranteed; it is quite possible that some sources with high data rate capture an unfair fraction of the channel. In this section we show how fairness can be enforced using a Bandwidth Balancing (BB) policy that was inspired by the DQDB fairness algorithm reported in [27] and by the follow-up paper that extended it to distributed bottleneck flow control for wireless networks [28]. To illustrate the BB approach, it will be useful to refer to Fig. 8(a) example: multiple sources and single sink. It is clear that each source should maintain an adequate number of sessions (and/or a number of pipelined packets on each session) in order to maximize aggregate throughput based on total load conditions. However, it is possible that some sources, because of their position or their traffic characteristics may transmit more than their fair share.The throughput of each source must be controlled so as to yield max-min fairness. For M-FAMA we follow the BB model reported in [28]. The main departure from [28] is to control not the fraction of bandwidth used by each sender in a contention domain, but the number of multiple sessions and pipelined packets issued by each source. Following the BB algorithm, in M-FAMA each source measures over a proper history window the residual (i.e., unused) bandwidth in the acoustic channel. If some sources are hidden from others, a central sink as depicted in Fig 8(a) can by default hear everyone and can thus propagate the residual bandwidth information. The residual capacity R_i at node i is expressed as:

$$R_i = \frac{T_{idle}}{T_p} C_i \tag{2}$$

C_i is the channel capacity at node i (same for all nodes in our case), T_p denotes the last measuring period and T_{idle} is the measured idle time. Large T_p allows accurate channel view but also long response time affecting the source node's ability to react to network changes. In this study, T_p was set to 10 × maximum propagation delay. The new allowed data rate $\gamma_i{}'$ at node i is then expressed as:

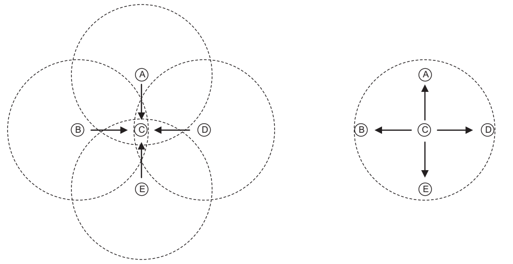
$$\gamma_i{}' = \alpha(R_i/OCC + \gamma_i) \tag{3}$$

Here, γ_i is the current sending rate. Each node only takes a fraction α of the available bandwidth. The coefficient α varies between [0, 1]. If α is small, a large amount of bandwidth is wasted but the network converges fast to a fair operating point. If α is close to 1, a small amount of bandwidth is wasted but the network convergence time increases. Besides the BB bandwidth constraint, another constraint on the number of packets/sessions is posed in M-FAMA by the overheard control packets and corresponding delay maps. For example, a source that currently has only one session and is allowed an extra session by BB, can start the new session only if that session will not interfere with scheduled transmissions from other sources. The BB control scheme proved to be very effective to manage congestion and fairness and was implemented in the M-FAMA simulator model. All the M-FAMA experiments incorporate BB, except otherwise specified.

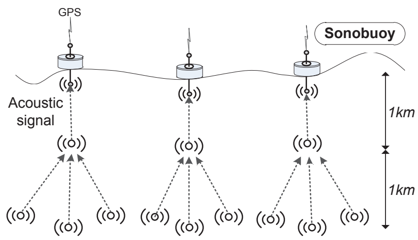
IV. Simulation & Evaluation

A. Simulation Setup

For acoustic communications, the channel model described in [29] and [30] is implemented in the QualNet physical layer. As in [29], [31], we use Rayleigh fading to model small-scale fading. Unless otherwise mentioned, the data rate is set to 16*kbps* as in [32], [33]. We use two different transmission ranges of 750*m* and 1500*m*. We measure throughput consumption per node as a function of the offered load per node. The load is varied from a single frame generated every



(a) 4-Senders 1-Sink (b) 1-Sender 4-Sinks



(c) Sea Swarm (tree)

Fig. 8. Simulation Topologies

30*sec* up to a single frame every 0.25*sec*. In some scenarios the offered load exceeds the available capacity. In such cases, the stream rate is adaptively reduced causing a reduction in video resolution or a decrease in the number of frames per second. Each simulation experiment was run for 1*hour*.

We start by examining M-FAMA's performance in the four topologies (see Fig. 8). The 4-sender/1-sink star topology as in Fig. 8(a) features four sender nodes competing to send their data to the center sink node. This topology is representative of a sea swarm engaged in scouting an underwater region and reporting multimedia information to a U/W command post. In the 1-sender/4-sink star topology (see Fig. 8(a)), the sender and receiver roles are reversed. This topology is representative of a reliable multimedia file transmission from a single source (say diver or AUV) to nearby divers or ships or other agents. The 4 sink configuration provides an opportunity to examine temporal reuse in action, since the source staggers the transmissions over time. The SEA Swarm (tree) topology represents a sensor data gathering scenario in which a swarm of sensors deliver data to the closest sonobuoy on the surface (with a static routing table). The final scenario is a dynamically varying topology in which the nodes are randomly placed in a 3D cube and are carried by underwater currents. The random deployment features ten fully-connected nodes, moving within the 3D cube based on the ocean current model called MCM [6].

We compare M-FAMA with well-known underwater CSMA protocols, namely Slotted FAMA (S-FAMA) [24], DACAP [34], and DOTS [21]. S-FAMA is a synchronized underwater MAC protocol that eliminates the need for excessively long control packets via time slotting. DACAP is a non-synchronized CSMA protocol that allows each node pair to use different RTS/CTS handshake intervals depending on distance between nodes. To cope with possible collision, DACAP requires both the sender and receiver nodes to send warning messages when they detect possible collision, thus deferring pending data reception/transmission. DOTS is a synchronized CSMA protocol that harnesses both temporal and spatial reuse to improve throughput. Like M-FAMA, DOTS relies on overheard neighboring node transmissions, but it lacks the support of multiple sessions from the source. The Bandwidth Balancing gain parameter for M-FAMA is set to α as 0.8.

Two M-FAMA variants are used in the simulation. As previously discussed, the variants differ in the allowed number of per-destination sessions for each sender node. Results for these variants are labeled as M-FAMA (*Con*) and M-FAMA (*Agg*), where *Con* is M-FAMA's Conservative mode and *Agg* is Aggressive mode. We measure the throughput of each flow at the receiver as a function of the offered load at each source as in [34], [21]:

$$\text{Offered Load} = \frac{\text{\# of generated } data \text{ frames} \times \text{Data size}}{\text{Simulation Duration} \times \text{Data rate}} \quad (4)$$

$$\text{Throughput} = \frac{\text{\# of rx } data \text{ frames} \times \text{Data size}}{\text{Simulation Duration} \times \text{Data rate}} \quad (5)$$

B. Simulation Results

1) 4-Sender/1-Sink Topology: Examining Fig. 9 and Fig. 10 we note that for range of 750*m* (Fig. 9), M-FAMA (Con) outperforms all the other protocols by large margins. The superior performance of M-FAMA (Con) over M-FAMA (Agg) results from the fact that four senders are competing for the channel over a short range, leading to congestion. M-FAMA (Agg) attempts to open multiple sessions and suffers from a high RTS collision rate. When the range is extended to 1500*m* as in Fig. 10 the load is reduced because of the higher round trip delay and M-FAMA (Agg) shows the best throughput outperforming M-FAMA (Con) by 10%. M-FAMA (Agg) takes advantage of the increased propagation delay, by pipelining a larger number of sessions.

Recalling that M-FAMA is equipped with BB control, we demonstrate in Fig. 11 and Fig. 12 how the BB technique works in asymmetric traffic situations that are prone to unfairness. The topology is the same as in Fig. 10. However, in Fig. 11, the sources start at different times. A and B start at time 0s while D and E start at time $90s$. As explained before, since packet size is fixed, the sender increases/decreases its rate by controlling the number of consecutive session it opens and thus packets it sends. In our example, senders A and B, upon the arrival of packets from D and E, follow the BB instructions by gradually decreasing the number of outstanding sessions. The Bandwidth Balancing algorithm converges after $120s$. At equilibrium, all senders have roughly the same relative throughput of 0.15 as already noted in Fig. 10. In the second experiment, two sources send a multimedia stream, and the remaining two sources send low rate sensor data (e.g., position, temperature, etc.). Max Min Fairness requires that the sensor sources transmit their full input rates, while the multimedia streams share the left over bandwidth. Fig. 12 shows exactly this behavior. The aggregate throughput is about 0.6 i.e., the same as in Fig. 10 where 4 multimedia sources are active. Fig. 13 confirms that BB is essential to provide fairness and maintain stability. The two M-FAMA version (Con) and (Agg) are tested with and without BB. The versions without BB collapse and can achieve only $1/3$ of the throughput of the BB version. Moreover, BB does not introducing significant signaling overhead. Recall that the tradeoff between convergence time and bandwidth efficiency is

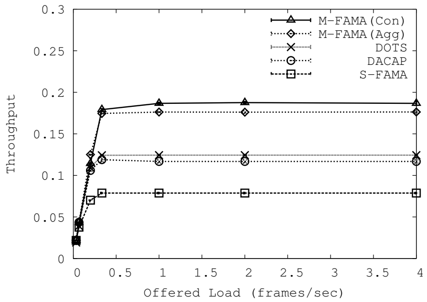


Fig. 9. 4Senders-1Sink: throughput as a function of offered load with tx range of *750m*

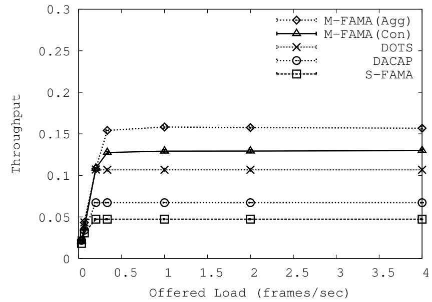


Fig. 10. 4Senders-1Sink: throughput as a function of offered load with tx range of *1.5km*

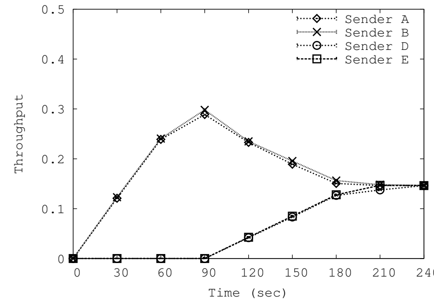


Fig. 11. 4Senders-1Sink: M-FAMA's throughput convergence under homogeneous traffic

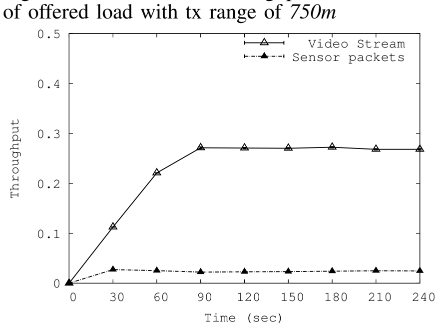


Fig. 12. 4Senders-1Sink: M-FAMA's throughput convergence under heterogeneous traffic

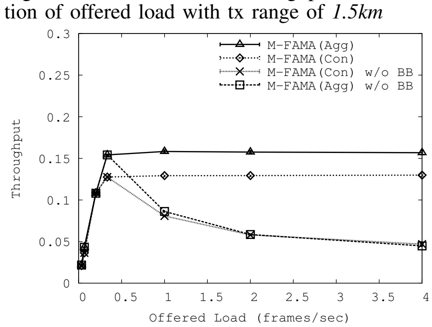


Fig. 13. 4Senders-1Sink:Bandwidth Balancing

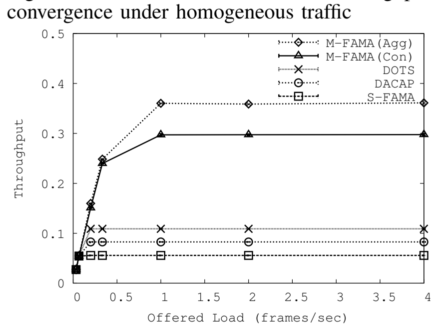


Fig. 14. 1Sender-4Sinks: throughput as a function of offered load with tx range of *750m*

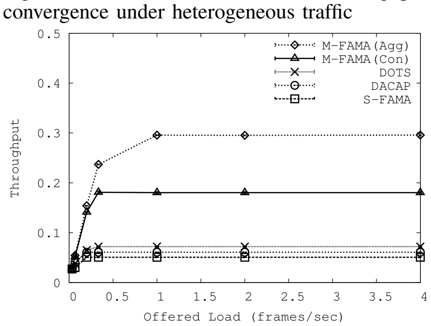


Fig. 15. 1Sender-4Sinks: throughput as a function of offered load with tx range of *1.5km*)

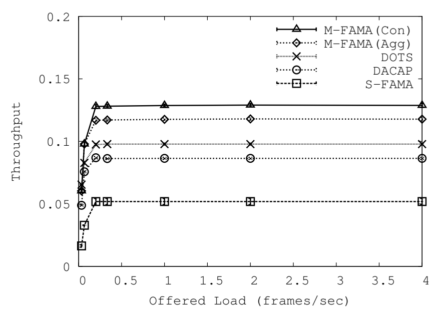


Fig. 16. Sea swarm (tree): throughput as a function of offered load

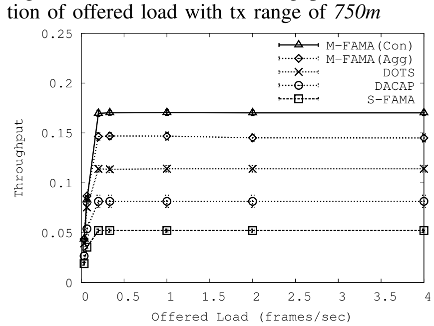


Fig. 17. Random with MCM (0.3m/s): throughput as a function of offered load

controlled by the gain parameter α. If the applications require faster convergence, this can be done by reducing the gain factor, at the expense of bandwidth efficiency.

2) 1-Sender/4-Sink Topology: Fig. 14 and Fig. 15 show the results. For both transmission range of 750*m* (Fig. 14) and 1.5*km* (Fig. 15), M-FAMA (Con) and M-FAMA (Agg) dominate the other protocols. For the load of 4 packets/sec, M-FAMA (Con) and M-FAMA (Agg) outperform the rest by 200–500%. This topology is the ideal scenario for demonstrating M-FAMA's ability for spatial and temporal reuse. Note that the central node has complete knowledge of all ongoing communication sessions (thus avoiding any collisions). As the number of session increases, the aggressive scheduling allows for higher throughput. If we compare this scenario with the previous scenario (4-sender/1-sink), we note that in the 4 to 1 scenario the outside nodes have limited knowledge of the other ongoing sessions. With this limited knowledge, they are more likely to initiate sessions that will result in RTS collisions at the central node. In contrast, in the 1 to 4 scenario, the center knows everything. This explains the stronger performance of M-FAMA in this scenario.

3) Sea Swarm (Tree) Topology: Fig. 16 shows the results for Sea Swarm depicted in Fig. 8(c). For simplicity, the topology is static, and all flows traverse from the bottom to the top (sonobouys). Now every packet is generated in the bottom nodes, and two hops are needed to deliver the packets to the surface. Conceptually, this scenario is similar to a multiple source/single sink scenario replicated over two hops. Not surprisingly the results are similar to those of the 4sender-1sink topology. The middle nodes suffer from congestion—there are three incoming flows, and a middle node is also in charge of transmitting data to the sonobouy. As expected, M-FAMA (Con) outperforms all the other protocols because of the heavy load and the relatively short range (1000*m*). Fair bandwidth sharing is also an important issue. Note that, in this topology, perfect bandwidth sharing is impossible due to high traffic concentration in the middle nodes. M-FAMA and DOTS exhibit about 0.8 of Jain's fairness index [35]. The other protocols fare worse, S-FAMA with index $= 0.5$ and DACAP with index $= 0.4$ respectively.

4) Random Topology with Meandering Current Mobility (MCM): The effects of random topologies and node mobility are examined in Fig. 17. Ten nodes are randomly deployed in a 3D cube with dimensions ($866m$ * $866m$ * $866m$). Given the node transmission range, this topology enables full connectivity among all nodes. Each node follows a jet stream path generated by the MCM model [6]. The main jet stream speed of each node is set to $0.3m/s$. The transmission range is set to 1.5*km*. Five sender-receiver node pairs are actively engaged in data communications, exchanging 128*byte* data packets. Fig. 17 shows that M-FAMA (Con) outperforms M-FAMA (Agg) by 15%, DOTS by 35%, DACAP by 100% and S-FAMA by 200%. M-FAMA (Con)'s superior performance over M-FAMA (Agg) is due to the relative high load caused by ten nodes in a relatively small cube, with low propagation delays.

5) Fairness: MAC protocols with backoff schemes based on incomplete information about network congestion may exhibit spatial unfairness (a form of channel capture), as described in Syed *et al.* [26]. We already addressed unfairness in M-FAMA and showed how to overcome it with the BB algorithm. We now examine the fairness of other protocols as well, using the benchmark topology in Fig. 17 (random with MCM). To measure fairness, we use the Jain Fairness Index [35].

$$\text{Fairness Index} = \frac{(\sum x_i)^2}{(n \cdot \sum x_i^2)} \quad (6)$$

The results are shown in Fig. 18. We already knew that M-FAMA + BB is fair. Fig. 18 confirms it. We also discover that DOTS exhibits a high fairness index (0.9 and above), This is explained by the fact that DOTS in this case works in a Round Robin mode. Each source transmits one packet when it gets its turn. S-FAMA and DACAP on the other hand have low Jain Index and are subject to severe unfairness and capture. The capture and unstable behavior is also revealed by the large fairness index variance, indicating that different sources manage to capture the channel at different load conditions. Moreover, these results indicate that the problem of unfairness and capture in underwater networks is severe. Techniques like Bandwidth Balancing must be used to reestablish stability and fairness.

6) Guard Time and Energy: We also explored the impact of nodal speed and M-FAMA's guard times, and the energy consumption of various protocols. Due to limited space, we only provide a brief summary in the following. First, to understand the impact of speed, we varied the maximum speed of nodes (i.e., AUVs) from $0.3m/s$ to $3m/s$, and found that mobility does not cause any significant throughput changes to M-FAMA under the scenarios considered. Second, we evaluated M-FAMA's performance by varying the guard time in a mobile scenario (with $0.3m/s$). Recall that if the guard time is too short, packet collisions significantly reduce throughput. If the guard time is too long, the lower temporal/spatial reuse reduces throughput. We found that the 1*ms*, 2*ms*, and 4*ms* guard time intervals achieve nearly identical throughput, while the 8*ms* and 16*ms* times have lower throughput due to lower utilization. In terms of energy consumption, we found that M-FAMA (Con) consumes more energy than S-FAMA, DACAP and DOTS because it delivers far more control frames than these three protocols. However, the per-DATA energy for M-FAMA (Con) is significantly lower than the other protocols. In the dense random scenario, M-FAMA (Agg) consumes more energy than M-FAMA (Con) even though it delivers fewer frames because the high node concentration leads to greater channel contention.

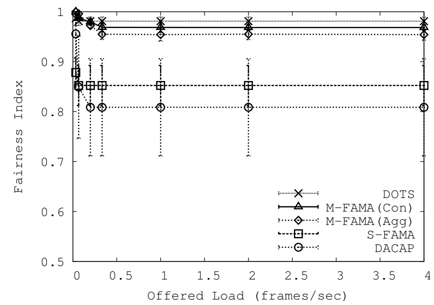


Fig. 18. Jain's Fairness Index for the five protocols

V. Related Work

Yackoski *et al.* [36] proposed UW-FLASHR, a variant TDMA protocol that can achieve higher channel utilization than existing TDMA protocols. Hsu *et al.* [12] proposed ST-MAC, an underwater TDMA protocol that operates by constructing a Spatial-Temporal Conflict Graph (ST-CG) to describe the conflict delays among transmission links, and reduces the ST-CS model to a new vertex coloring problem. A heuristic, called the Traffic-based One-step Trial Approach (TOTA), is then proposed to solve the coloring problem. Kredo *et al.* [13] proposed a TDMA-like protocol called STUMP that uses propagation delay information and prioritizes conflicting packet transmissions based on certain metrics (e.g., random ordering and uplink delay ordering). Ma *et al.* [37] proposed an efficient scheduling algorithm with constant approximation ratios to the optimum solutions for both unified and weighted traffic load scenarios. However, TDMA-like protocols are not well suited to mobile, resource-constrained sensor networks due to poor failure resilience, computation effort, and requirement for network-wide time consensus.

CSMA-like protocols (or reservation-based protocols) have also been proposed to exploit temporal reuse in several ways. Given that channel reservation (i.e., RTS/CTS) takes a long time, Guo *et al.* proposed Adaptive Propagation-delay-tolerant Collision Avoidance Protocol (APCAP). This scheme allows a node to transmit packets out-of-order [17], but does not detail scheduling strategies for out-of-order packet delivery. To reduce the control overhead (e.g., reservation, acknowledgement), R-MAC [18] delivers a burst of packets combined with delayed ACKs, thereby improving the channel throughput. Chen *et al.* proposed Ordered CSMA where the nodes transmit their data packets in a fixed order as in a Daisy Chain [19]. Given the fact that two sequential signals traveling in the same direction will not collide, each station transmits immediately after receiving a data frame from the previous station sequentially, instead of waiting for a period of maximum propagation delay. Yet, ordered CSMA is not appropriate for large-scale, mobile and multi-hop networks because generating an efficient node ordering requires knowledge of relative positions of all nodes in the network and a large number of packet exchanges. Moreover, it will be costly to maintain the order as nodes move and dynamically enter/exit the transmit chain.

Chirdchoo et al. [20] proposed a receiver-initiated reservation protocol called Receiver-Initiated Packet Train (RIPT) where after inviting neighbors with packets to transfer, the receiver accepts their transmission and builds a transmission schedule for the neighbors based on propagation delays. In

RIPT, the receivers need to periodically initiate the packet transfers; under varying traffic demands, it is non-trivial to determine when to initiate packet transmissions. Chirdchoo *et al.* [38] proposed another reservation based protocol, MACA-MN, which increases the channel utilization by enabling multiple packet trains to the neighbors. MACA-MN allows each sender to send packet trains to the multiple neighbors by transmitting RTS with embedded information of the number of DATA packets for multiple intended neighbors and the inter-node propagation delay from the sender to its intended receivers. Noh *et al.* proposed the DOTS protocol [21] which harnesses both temporal and spatial reuse to improve channel utilization. However, DOTS' ability of temporal and spatial reuse is limited to the receiver side. There is no support for a sender to open concurrent sessions to the same destination. Unlike existing underwater CSMA solutions, M-FAMA neither requires an additional phase for reservation scheduling nor restricts transmission schedules to a specific order. M-FAMA is a sender-initiated protocol that relies solely on passively-overheard neighboring transmissions to make intelligent local scheduling of multiple concurrent sessions.

VI. Conclusions

M-FAMA is a MAC protocol designed for underwater acoustic streams. It allows a source to open multiple sessions to different receivers achieving temporal/spatial reuse and yet avoiding collisions by careful accounting of neighbors' transmission schedules. It supports packet pipelining on the same link, with significant throughput improvement when only one node pair is active. M-FAMA is a greedy protocol that attempts to maximize throughput at the expense of fairness. To correct this tendency a fully distributed, low overhead Bandwidth Balancing control was introduced that guarantees stability and fairness in arbitrary topologies and traffic patterns. To further prevent congestion, a conservative version of M-FAMA was introduced, that restricts pipelining, with promising results in dense, heavy loaded networks with low propagation delays. Extensive simulation experiments have shown that M-FAMA outperforms the most popular underwater MAC protocols in representative streaming scenarios. Future work will include the study of a dynamic M-FAMA protocol that switches to conservative mode in specific load and topology conditions, and; the extension of M-FAMA to multicast applications and opportunistic forwarding.

References

[1] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater Acoustic Sensor Networks: Research Challenges," *in Ad Hoc Networks (Elsevier)*, 2005.

[2] J. Kong, J. H. Cui, D. Wu, and M. Gerla, "Building Underwater Ad-hoc Networks and Sensor Networks for Large Scale Real-time Aquatic Applications," in *MILCOM*, 2005.

[3] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke, "Data Collection, Storage, and Retrieval with an Underwater Network," in *Sensys*, 2005.

[4] U. Lee, J. Kong, J. S. Park, E. Magistretti, and M. Gerla, "Time-Critical Underwater Sensor Diffusion with No Proactive Exchanges and Negligible Reactive Floods," in *ISCC*, 2006.

[5] Z. Zhou, J. Cui, and A. Bagtzoglou, "Scalable Localization with Mobility Prediction for Underwater Sensor Networks," in *INFOCOM*, 2008.

[6] A. Caruso, F. Paparella, L. Vieira, M. Erol, and M. Gerla, "The Meandering Current Mobility Model and its Impact on Underwater Mobile Sensor Networks," in *Infocom*, 2008.

[7] R. J. Urick, "Principles of Underwater Sound 3rd Edition," *McGraw-Hill*, 1983.

[8] "WHOI, *Micro-Modem Overview*, Woods Hole Oceanographic Institution," http://acommn.whoi.edu/micromodem.

[9] J. Partan, J. Kurose, and B. N. Levine, "A Survey of Practical issues in Underwater Networks," in *WUWNet'06*, September 2006.

[10] E. Sozer, M. Stojanovic, and J. Proakis, "Underwater Acoustic Networks," *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 72–83, Jan. 2000.

[11] J. Preisig, "Acoustics Propagation Considerations for Acoustics Communications Network Development," in *ACM WUWNet'06*, 2006.

[12] C. Hsu, K. Lai, C. Chou, and K. C. Lin, "ST-MAC: Spatial-Temporal MAC Scheduling for Underwater Sensor Networks," in *INFOCOM*, 2008.

[13] K. Kredo, P. Djukic, and P. Mohapatra, "STUMP: Exploiting Position Diversity in the Staggered TDMA Underwater MAC Protocol," in *INFOCOM*, 2009.

[14] J. Rice, "SeaWeb Acoustic Communication and Navigation Networks," in *Intl. Conf. Underwater Acoustic Measurements*, 2005.

[15] N. Baldo, P. Casari, and M. Zorzi, "Cognitive Spectrum Access for Underwater Acoustic Communications," in *ICC Workshops*, 2008.

[16] N. Baldo, P. Casari, P. Casciaro, and M. Zorzi, "Effective Heuristics for Flexible Spectrum Access in Underwater Acoustic Networks," in *OCEANS*, 2008.

[17] X. Guo, M. Frater, and M. Ryan, "A Propagation-delay-tolerant Collision Avoidance Protocol for Underwater Acoustic Sensor Networks," in *OCEANS*, 2006.

[18] P. Xie and J. H. Cui, "R-MAC An Energy-Efficient MAC Protocol for Underwater Sensor Networks," in *WASA*, 2007.

[19] M. Chen, S. Gonzalez, and V. Leung, "Applications and Design Issues for Mobile Agents in Wireless Sensor Networks," *Wireless Communications, IEEE*, 2007.

[20] N. Chirdchoo, W. seng Soh, and K. Chua, "RIPT: A Receiver-Initiated Reservation-Based Protocol for Underwater Acoustic Networks," *Selected Areas in Communications, IEEE Journal on*, 2008.

[21] Y. Noh, P. Wang, U. Lee, D. Torres, and M. Gerla, "DOTS: A Propagation Delay-aware Opportunistic MAC Protocol for Underwater Sensor Networks," in *ICNP*, 2010.

[22] P. Karn, "MACA : A New Channel Access Protocol for Packet Radio," in *ARRL/CRRL*, September 1990.

[23] C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Solutions to Hidden Terminal Problems in Wireless Networks," in *SIGCOMM*, 1997.

[24] M. Molins and M. Stojanovic, "Slotted FAMA: A MAC Protocol for Underwater Acoustic Networks," in *OCEANS*, 2006.

[25] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: a Media Access Protocol for Wireless LAN's," in *SIGCOMM*, 1994.

[26] A. A. Syed, W. Ye, and J. Heidemann, "T-Lohi: A New Class of MAC Protocols for Underwater Acoustic Sensor Networks," in *Infocom*, 2008.

[27] E. Hahne, A. Choudhury, and N. Maxemchuk, "DQDB Networks with and without Bandwidth Balancing," *Communications, IEEE Transactions on*, vol. 40, no. 7, pp. 1192 –1204, Jul 1992.

[28] C. Zhou and N. Maxemchuk, "Bandwidth Balancing in Mobile Ad hoc Networks," in *WiMob*, oct. 2010, pp. 15 –20.

[29] M. Stojanovic, "On the relationship between capacity and distance in an underwater acoustic communication channel," *SIGMOBILE Mob. Comput. Commun. Rev.*, 2007.

[30] P. L. L.M. Brekhovskikh, Yu, *Fundamentals of Ocean Acoustics 3rd Edition*. Springer, 2003.

[31] C. Carbonelli and U. Mitra, "Cooperative Multihop Communication for Underwater Acoustic Networks," in *WUWNet*, 2006.

[32] Z. Zhou and J.-H. Cui, "Energy Efficient Multi-path Communication for Time-critical Applications in Underwater Sensor Networks," in *MobiHoc*, 2008.

[33] U. Lee, P. Wang, Y. Noh, L. F. M. Vieira, M. Gerla, and J.-H. Cui, "Pressure Routing for Underwater Sensor Networks," in *INFOCOM*, 2010.

[34] B. Peleato and M. Stojanovic, "Distance Aware Collision Avoidance Protocol for Ad-hoc Underwater Acoustic Sensor Networks," *Communications Letters, IEEE*, 2007.

[35] R. Jain, D. Chiu, and W. Hawe, "A Quantitative Measure Of Fairness And Discrimination For Resource Allocation In Shared Computer Systems," DEC Research, Tech. Rep., 1984.

[36] J. Yackoski and C.-C. Shen, "UW-FLASHR: Achieving High Channel Utilization in a Time-based Acoustic MAC Protocol," in *WUWNet*, 2008.

[37] J. Ma and W. Lou, "Interference-aware Spatio-temporal Link Scheduling for Long Delay Underwater Sensor Networks," in *SECON*, 2011.

[38] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "MACA-MN: A MACA-Based MAC Protocol for Underwater Acoustic Networks with Packet Train for Multiple Neighbors," in *VTC Spring 2008. IEEE*.