Fair and Throughput-Optimal Routing in Multimodal Underwater Networks

Roee Diamant, *Member, IEEE*, Paolo Casari, *Senior Member, IEEE*, Filippo Campagnaro, *Student Member, IEEE*, Oleksiy Kebkal, Veronika Kebkal, Michele Zorzi, *Fellow, IEEE*

Abstract—While acoustic communications are still considered the most prominent technology to communicate under water, other technologies are being developed based, e.g., on optical and radio-frequency electromagnetic waves. Each technology has its own advantages and drawbacks: for example, acoustic signals achieve long communication ranges at order-of-kbit/s bit rate, whereas optical signals offer order-of-Mbit/s transmission rates, but only over short ranges. Such a diversity can be leveraged by multimodal systems, which integrate different technologies and provide the intelligence required to decide which one should be used at any given time. In this paper, we address a fundamental part of this intelligence by proposing Optimal Multimodal Routing (OMR), a novel routing protocol for underwater networks of multimodal nodes. OMR makes distributed decisions about the flow in each link and over each technology at any given time, in order to advance a packet towards its destination; in doing so, it prevents bottlenecks and allocates resources fairly to different nodes. We analyze the performance of OMR via simulations and in a field experiment. The results show that OMR successfully leverages all technologies to deliver data, even in the presence of imperfect topology information. To permit the reproduction of our results, we share our simulation code.

Index Terms—Underwater networks; underwater acoustic communications; optical communications; multimodal systems; optimal routing; simulations; lake trial

I. INTRODUCTION

Several different physical layer (PHY) technologies have been developed to communicate under water. While most of them rely on acoustic communications at different frequencies and over different bandwidths [2], optical communications are also gaining momentum [3], [4], [5], and recent work suggests that radio-frequency (RF) electro-magnetic communications are also finding their way into research interests [6] and system development [7]. Novel system architectures based on electro-static fields [8] and magneto-inductive communications [9] are

also being explored, albeit these technologies are still in their infancy compared to acoustics, optics, and electromagnetics.

Each of the above underwater PHY technologies offers a different balance of advantages and disadvantages. The most prominent differences between underwater acoustic and optical communications, for instance, concern the data rate and the communications range. Acoustic modems typically provide low (order-of-kbit/s) transmission rates, as their performance is highly influenced by the environment, especially by strong and time-varying multipath [10]. In turn, recent advances in acoustic physical layer [2] and modem hardware/software design [11] allow acoustic devices to cover ranges up to several kilometers in low-frequency systems, and to achieve bit rates of the order of several tens of kbit/s for high-frequency, short-range systems. Underwater optical communications, on the other hand, provide a very high data rate, up to tens of Mbit/s, but require the transmitter and receiver to be close, up to a few meters apart in typical shallow water scenarios. In addition, optical transceivers typically have to be aligned within each other's field of view in order to establish a link. In addition, optical communications are sensitive to turbidity and tend to work best in dark waters. By way of contrast, RF communications do not need any alignment and can be developed based on very standard hardware already used for terrestrial radio systems; however, the conductivity of ocean waters attenuates RF waves within very short distances, and limits the achievable bit rates to less than 100 kbit/s within a distance of a few tens of m [6].

The above analysis suggests that the integration of different PHYs into a multimodal communication system holds a lot of potential. A system is defined to be multimodal when it encompasses any set of non-mutually interfering technologies. For instance, a multimodal system can rely on acoustic and optical technologies, or even on a set of acoustic communication technologies working on different bands. Such a system may be able to exploit the advantages of different technologies by transmitting through the best available one at any given moment. This approach was proposed, e.g., in [12]. After comparing the declared performance of the technologies available at the time, the authors concluded that a system encompassing optical and acoustic communications would be a good candidate for the wireless control of remotely operated vehicles. Notably, recent work [13] supports the vision of multimodal systems by showing that embedded processing platforms have sufficiently evolved to host the signal processing algorithms of acoustic communication systems on general-purpose computing platforms. This means that both

the complexity and the versatility of these systems are already borne easily by current hardware, making multimodal communications *de facto* possible already with current technology.

A key role, in multimodal communication systems, is played by the logic that decides how to switch between the available PHYs. While multimodal point-to-point links are manageable with relatively simple policies [14], organizing multimodal nodes into a network requires a change of perspective. In fact, the nodes may connect to different neighbors using (possibly partially overlapping) subsets of their PHYs. These subsets may change over time according to a variety of circumstances, that depend, e.g., on environmental conditions, mobility, and on the traffic requirements of the nodes. In this paper, we design a specific component of the multimodal PHY usage logic: the multihop routing algorithm. In this context, we propose Optimal Multimodal Routing (OMR), a routing solution that fully utilizes the available PHY technologies in an optimized fashion. Specifically, by considering the different PHY technologies as another layer of network resources, we formalize the routing problem as a maximization problem where each node tries to extract the most from all its available PHYs. The solution to this problem leads to a routing protocol that is distributed and fair, and avoids bottlenecks. OMR is valid in any network topology, and can be applied to any combination of available PHY technologies, including when different nodes incorporate different technologies.

Our contribution is twofold, and includes:

- A novel distributed routing algorithm for multimodal underwater networks, which maximizes the amount of information transmitted through all technologies available to each node, while at the same time balancing the traffic flow through the network and pursuing a fair network utilization for all nodes;
- A broad quantitative evaluation, which proves that our distributed algorithm performs very close to a globally optimum benchmark relying on full network topology knowledge, and that it outperforms both straightforward solutions to exploit multimodal links and more complex multi-technology routing solutions inherited from the terrestrial radio networking domain.

We evaluate the performance of OMR by means of numerical simulations and through several field experiments performed in a lake north of Berlin, Germany. To the best of our knowledge, ours is the first reported trial for multimodal routing schemes.

The remainder of this paper is organized as follows. Section II discusses related work; Section III presents and formalizes our routing algorithm; Section IV describes our simulation scenario and discusses the simulation results; Section V reports the results of the field experiments; Section VI draws concluding remarks.

II. RELATED WORK

"Multimodal" is a relatively recent term that refers to communication solutions encompassing two or more PHY subsystems, based either on different technologies (e.g., acoustic and optical) or on different implementations of the same PHY technology (e.g., on multiple acoustic systems working in different, non-mutually interfering frequency bands). However, an implementation of a multimodal system with straightforward switching policies was already presented in [15], where the authors designed a data mule AUV that should approach each node of a deployed underwater acoustic network and retrieve data using optical communications. Acoustic and optical communications are typical technologies employed in multimodal systems. As shown by the survey in [12], radio frequency technologies for underwater communications are also under development, but their declared performance is topped by that of optical and acoustic systems at all distances of interest for RF technologies. In particular, the considerable maturity achieved by acoustic systems yields bit rates on the order of tens of kbit/s over fairly long distances [16], and the higher-than-Mbit/s rates of optical modems in sufficiently benign waters are still unrivaled [3]. A notable feature of multimodal systems is that the composition of multiple powerful PHYs may not be necessary to achieve good performance. Indeed [17] demonstrates that even a low-bit rate, minimal-cost infrared optical modem, assembled starting from very inexpensive parts, can substantially improve the performance of underwater acoustic networks. The authors employed time synchronization and TCP connections as use cases.

The variable-depth moored nodes presented in [18] join acoustic communications under water and radio communications on top of the water surface. The autonomous underwater exploration platform discussed in [19] is multimodal in the sense that it can rely on multiple sensors, and has different underwater communication capabilities. Specifically, the authors discuss the tradeoffs between frequency-shift keying (FSK)-based modem technology and custom low-cost modems designed around a commercial off-the-shelf ceramic transducer [20]. The work in [21] employs multimodal optical and acoustic communications in a clustered optical underwater network. Long range acoustics are exploited for cluster formation and management, whereas intra-cluster communications take place through optical connections.

Hybrid acoustic/optical multimodal networks are considered for the transmission of real-time video streams in [22]. Bulk data streaming takes place through the optical channel, whereas acoustic communications are leveraged to send acknowledgments, to transmit data during the alignment of optical modems, and as a fallback solution in turbid waters. In [23], the authors assume that sensor data generated by underwater nodes lose value over time. The path of an autonomous underwater vehicle (AUV) is then optimized to maximize the value of the information from the sensors. Sensor-to-AUV data upload takes place through an optical connection, whereas the sensors notify the AUV of new data using control packets through acoustic connections. Context-based switching schemes are considered in [12] to manage multimodal optical and acoustic communications for the wireless remote control of a remotely operated vehicle (ROV). By tuning the parameters of the switching policies, it is shown how the ROV-controller link can benefit from each technology and how well the ROV reacts to the controller's commands. This work was extended with the design of proactive switching policies in [14]. In [24], more complex scenarios are implemented

using the free-access DESERT underwater framework [25] and evaluated in a diver cooperation scenario.

The authors in [4] propose hybrid acoustic/optical communications to coordinate swarms of autonomous underwater vehicles and to transfer information among swarm components. The custom design of both the acoustic and the optical modem is also discussed. A bilingual modem concept was implemented in [26] using a custom re-configurable underwater acoustic modem. JANUS was employed both for first contact and as a robust fallback modulation scheme, whereas the native FSK scheme of the acoustic modem was resorted to under sufficiently benign channel conditions.

While the above works were key to introduce and improve the use of multimodal technologies in underwater networks, the approaches taken are often not formal, and rely on heuristic solutions to divide the data between the PHY technologies. As a result, the multimodal network is not optimally utilized. Considering this challenge, in this paper we present a routing algorithm that aims at maximizing the multimodal network goodput under constraints that divide resources fairly among the nodes, and avoid that some nodes become bottlenecks for the routing process.

The optimal assignment of data traffic to different communication technologies in a multimodal underwater network is related to the design of multipath routing schemes in multihop multi-channel networks [27]. The motivation behind these protocols is typically to exploit additional network resources to improve reliability, increase throughput or achieve load balancing across different sections of a wireless network. The problem implies several challenges related to interface assignment and switching, routing metric design, and mobility [28]. In the literature on terrestrial radio networks, few works design protocols to optimally assign a data flow to a specific path in a network of multi-channel nodes [29], [30], [31].

Among basic schemes, the CA-AODV [29] protocol proposes to combine channel assignment with AODV [32] using one channel to exchange route requests and reply packets. Thus, the route discovery mechanism automatically finds the information required to make flow assignment decisions. Hyacinth [30] is a protocol for wireless mesh networks that spans trees from gateway nodes and routes data based on either hop count, gateway link capacity or minimum residual bandwidth over a given path. The multi-radio link quality source routing protocol chooses forwarding paths via an expected transmission time metric [31]. A very similar approach is employed in [33] to find multipath routes and optimally allocate video data flows over a wireless mesh network.

The terrestrial radio approaches above focus on relatively stable wireless networks and on the optimization of link reliability or load balancing for large networks. Unlike these approaches, underwater networks are commonly small and have a limited number of hops. The main challenge with underwater multimodal routing is how to combine transmissions over the technologies available to a node in the settings of poor physical layer reliability, where both transmissions and acknowledgments often fail.

The approach and metric proposed in [31] are often considered in the literature as a benchmark for performance evaluation of multipath, multichannel routing protocols (e.g., see [30], [33]). We will also consider a version of the method in [31] as a benchmark in our simulation study, as detailed in Section IV-A.

III. Network model and Optimal Routing

We consider a converge-casting network of multimodal underwater nodes, where data traffic has to be routed towards a common sink node. This is a very common network topology arrangement, that is found in several practical applications, from long-term data monitoring in a given area, to underwater infrastructure monitoring [34] and to Internet-accessible underwater networks [35]. We desire to obtain good performance in all key aspects of the multimodal network. In particular, we are interested in maximizing the network goodput while limiting the end-to-end transmission delay. Since underwater networks usually face energy limitations, we are also interested in minimizing the packet transmission overhead. On the other hand, to keep network traffic flow smoothly and reduce bottlenecks, we are interested in the full exploitation of the available network links. With the goal of obtaining a favorable tradeoff among the above quality measures, in this paper we propose the first optimal multimodal routing (OMR) scheme for underwater networks.

A. Key idea

The key idea behind our distributed routing scheme is that the available multimodal links should be fully exploited, while at the same time avoiding that some relays become bottlenecks for the routing process. To do so, the nodes should i) avoid forwarding an excessive amount of traffic towards the relays upstream and ii) favor nodes with fewer valid routes to the sink during the data relaying process. We achieve this by having the nodes estimate the capability of their relays to forward traffic further, and by having these relays distribute a minimal amount of information about the current backlog of data bits in their queues. This allows each network node to separately solve an optimization problem, and to find the number of bits to be transmitted to its neighbors through each multimodal link. We note that our approach does not resort to flooding, as we explicitly want to avoid unnecessary redundant transmissions of the same data.

Even without topology information, this approach balances traffic much better with respect to a baseline algorithm that, e.g., floods all data through all available technologies. This can be observed in Fig. 1, where we report the per-link transmission rates required to convey the same number of packets to the sink. We consider three multimodal routing solutions: flooding, where all links and all available technologies are used and packets are re-transmitted; our OMR method to achieve fairness with only one-hop topology information; and a version of our OMR method where the full topology information is available (OMR–FF), as a benchmark. OMR is a fully distributed protocol, while OMR–FF is centralized. Three acoustic PHY technologies are used by the network nodes: a low-frequency, low-rate technology (LF), a faster technology working at intermediate frequency (MF) and a
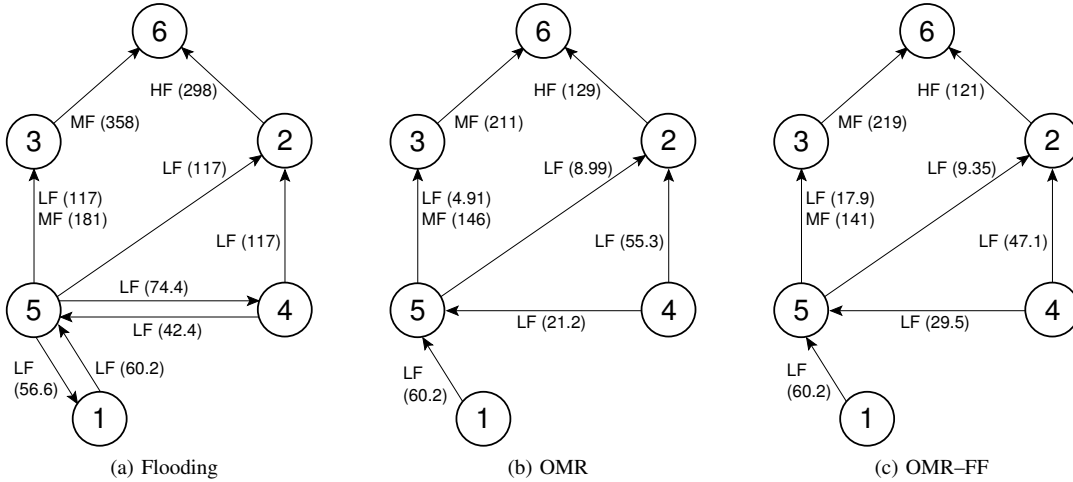
Fig. 1. Per-link transmission rates in bytes/s using flooding (a) and our OMR algorithm (b). OMR is also compared with the OMR–FF benchmark (c), where the full topology is assumed to be known at no cost.

high-frequency high-rate technology (HF). In parentheses, we show the obtained transmission rates in bytes/s. We remark that there is no conceptual limit to the number of technologies that a node can incorporate: we consider three technologies in this example in order to fix ideas.

From Fig. 1 we observe that, with respect to the flooding case (Fig. 1a), the transmission rates are much less intense and more balanced with our approach (Fig. 1b/c). In the ideal case of OMR–FF where full topology information is available (e.g., because the topology has been tested in advance, and does not change over time), the algorithm can better organize transmissions across all links compared to OMR. For example, OMR–FF redistributes part of the traffic of node 4 through node 5, resulting in better utilization of the LF link from node 3 to node 6, as can be seen by comparing Figs. 1b and 1c. In the following, we describe the routing algorithm in detail.

B. Preliminary definitions and assumptions

We assume that our underwater network is composed of a set $\mathcal{N}$ of multimodal nodes, where $|\mathcal{N}| = N$. The network implements a converge-casting scenario, where all nodes send their information over multiple hops to a common sink (denoted as D, e.g., node 6 in Fig. 1). We assume that the network topology has been already discovered.[1] While the implementation of a topology discovery algorithm is outside the scope of this paper, we do assume that the process can be subject to errors, or to inaccuracies caused by slow topology changes over time. We will take these errors into account in the design of the routing protocol. Given the outcome of the routing structure discovery, we assume that each node knows the available alternatives to forward a packet towards the sink D. Accordingly, for each node, we call $\mathcal{Y}_i$ the set of upstream neighbors of i, i.e., $\mathcal{Y}_i$ contains all one-hop neighbors of i that

can advance packets one further hop towards D (for example, $\mathcal{Y}_5 = \{2, 3\}$ in Fig. 1).[2] We also call $\widetilde{\mathcal{Y}}_i$ the list of *all* one-hop neighbors of i.

Each multimodal node incorporates a number of PHY technologies, listed in the set $\mathcal{T}_i$ (e.g. $\mathcal{T}_5 = \{\text{LF}, \text{MF}\}$ in Fig. 1). A node can communicate using any subset of technologies available to it and to the addressed receiver. Note that the list of available technologies may vary over time, e.g., due to channel variations or mobility. Let $\mathcal{T}_{i,j}^\tau$ be the set of technologies that i can use to transmit to j at time τ. We assume that this set of technologies is known to the routing protocol, e.g., because some underlying MAC protocol forwards a notification when a given technology is available. This process is outside the scope of this paper, and can be implemented, e.g., through the schemes in [14], [24].

Each node maintains a queue with a list of packets to transmit. Denote the bits in node i's queue at time τ as $P_{i,\tau}$. Define $R_i^\tau(j,t)$ as the number of bits in $P_{i,\tau}$ that will be sent by node i to node $j \in \mathcal{Y}_i$ using technology t at time τ. The objective of the routing algorithm is to find optimal values for $R_i^\tau(j,t)$, under a constraint on the total number of bits that can be transmitted by i using technology t over a user-defined time span u, denoted as $C_{i,t}^{(u)}$. We will indicate these optimal values as $\widehat{R}_i^\tau(j,t)$. A summary of the employed notation is provided in Table I. The table also reports the inter-dependencies among the quantities introduced above, and the nodes each quantity is shared with. This is meant as a reference for the algorithm description below.

The optimization is to be carried out using the information available at node i or passed on by its upstream neighbors. In particular, we assume that node i knows: $\mathcal{Y}_i$ and $\mathcal{Y}_j \ \forall j \in \mathcal{Y}_i$; $P_{i,\tau}$ and $P_{j,\tau'} \ \forall j \in \mathcal{Y}_i$, where $\tau' < \tau$ is a time epoch that refers to a transmission carried out by node j immediately preceding the current epoch τ; and $C_{j,t}^{(u)} \ \forall j \in \mathcal{Y}_i$.

---

[1]This can be done, e.g., by sending beacon packets downstream from the sink to the network nodes [36], or by carrying out processes aimed at discovering either the topology itself [37] or at least the available routes [38]. The discovered structure can be maintained by tracking transmission successes over each link over time [39]. These processes are out of the scope of this work.

[2]We remark that routing in the network is never performed downstream, i.e., no relaying operation will bring a packet one hop farther from the destination.

TABLE I
EXPLANATION OF THE EMPLOYED NOTATION

Symbol	Meaning	Requires	Shared with
$\mathcal{N}$	Set of the network nodes	—	—
D	Destination node	Known by all nodes	—
$\mathcal{Y}_i$	Set of upstream neighbors of node i	—	One-hop neighbors
$\widetilde{\mathcal{Y}}_i$	Set of all one-hop neighbors of node i	—	One-hop neighbors
$\mathcal{T}_i$	Set of PHY technologies available at node i	—	—
$\mathcal{T}_{i,j}^{\tau}$	Set of PHY technologies through which i can transmit to j at time τ	Technology availability signaled by MAC protocol	—
$P_{j,\tau}$	Number of bits in node j's queue at time τ	$P_{j,\tau'}$ for $\tau' < \tau$, $\widehat{R}_j^{\tau}(k,t)$ $\forall k \in \mathcal{Y}_j$	One-hop neighbors
$C_{i,t}^{(u)}$	Total number of bits that can be transmitted by i using technology t over a time period of duration u	Technology availability signaled by MAC protocol	—
$L_j^{(i)}$	Number of node-disjoint routes towards the destination D available to i when routing through j	Topology information or $\mathcal{Y}_\ell \; \forall \ell \in \mathcal{Y}_i \cup \widetilde{\mathcal{Y}}_i$	—
$F_j(i)$	Fair share of j's upstream transmission resources that can be dedicated to node i	$L_j^{(i)}$, $\mathcal{Y}_\ell \; \forall \ell \in \mathcal{Y}_i \cup \widetilde{\mathcal{Y}}_i$	—
$R_i^{\tau}(j,t)$	Number of bits in node i's queue sent to node j using technology t at time τ	$P_{i,\tau}$, Δ_j^{τ}, $C_{i,\tau}^{(u)}$, $F_j(i)$	—
Δ_j^{τ}	Amount of upstream transmission resources of node j that can be secured for node i's transmissions	$C_{j,t}^{(u)} \forall t \in \mathcal{T}_{j,k}^{\tau}$, $R_j^{\tau}(k,t)$ $\forall k \in \mathcal{Y}_j$	Nodes ℓ s.t. $i \in \mathcal{Y}_\ell$

C. Routing algorithm

We are now ready to describe the steps of the routing optimization algorithm executed by node $i \in \mathcal{N}$. Node i has to decide how many bits to transmit through each of its available technologies, and carries out the following steps for each upstream neighbor in $\mathcal{Y}_i$. For clarity, we will illustrate the algorithm by referring to one of these upstream nodes, j. The optimal transmitted bit allocation for node i is obtained by solving the following problem:

$$\widehat{R}_i^{\tau}(j,t) = \arg\max_{R_i^{\tau}(j,t)} \sum_{j \in \mathcal{Y}_i} \sum_{t \in \mathcal{T}_{i,j}^{\tau}} R_i^{\tau}(j,t) \qquad (1a)$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{Y}_i} \sum_{t \in \mathcal{T}_{i,j}^{\tau}} R_i^{\tau}(j,t) \leq P_{i,\tau} ; \qquad (1b)$$

$$\sum_{t \in \mathcal{T}_{i,j}^{\tau}} R_i^{\tau}(j,t) \leq \Delta_j^{\tau} ; \qquad (1c)$$

$$R_i^{\tau}(j,t) \leq C_{i,t}^{(u)} F_j(i) . \qquad (1d)$$

Constraint (1b) means that the bits transmitted across all technologies shall not exceed the remaining number of bits in the queue at node i. Constraint (1c) takes into account that i's upstream neighbor j may have a backlog of packets to be transmitted, and that node j would give priority to these bits in a FIFO fashion. Assuming that the remaining portion of j's upstream transmission resources after the transmission of the backlog is sufficient to transmit Δ_j^{τ} bits from node i, constraint (1c) makes sure that i transmits no more than Δ_j^{τ} bits to j, aggregate over all technologies. Finally, constraint (1d) means that the number of bits transmitted through either technology should not exceed a certain limit, defined as

i's fair share of j's upstream link capacity, where $F_\ell(j) = 0$ if node ℓ has nothing to transmit.

Note that, for a given node j, $\sum_i F_j(i)$ can exceed 1. This is because condition (1d) only applies if a node i has more possible relays to the sink than node j: in this case, it should divide its transmissions while considering the relay options of other neighbors of j. Also note that since we limit ourselves to a distributed solution, node i has typically no way to ascertain the technology used over link $j \to k$, $k \in \mathcal{Y}_j$. Instead, we perform technology allocation only hop-by-hop. As a result, the term $F_j(i)$ is not related to the used technology t.

The quantities required to evaluate the constraints are fully determined by node i. Node i is assumed to know the capacity of its one-hop links, its available technologies, and its different paths to the sink. However, Δ_j^{τ} and $F_j(i)$ must still be computed, as will be detailed in the following.

1) Calculation of the fair share of node j's resources:
We start with the computation of $F_j(i)$. The upstream transmission resources of node j are assigned to a downstream neighbor i depending on the number of node-disjoint routes towards the destination D available to i, indicated with $L_j^{(i)}$, where the subscript j indicates that $j \in \mathcal{Y}_i$, and that it is being considered as a next hop. The rationale behind the resource assignment strategy is that if some downstream neighbors m of j can reach the destination only via a route that passes through node j, such nodes m should be given a higher priority in the use of j's upstream transmission resources. Formally, define

$$\widetilde{L}_j^{(i)} = \sum_{\ell \in \mathcal{Y}_j} L_j^{(\ell)} - L_j^{(i)} . \qquad (2)$$

If $\widetilde{L}_j^{(i)} = 0$, then we immediately set $F_j(i) = 1$, as j is the only neighbor of i that can relay packets towards D (e.g.,

$F_5(1) = 1$ in Fig. 1). Otherwise, $F_j(i)$ is computed as

$$F_j(i) = \frac{\widetilde{L}_j^{(i)}}{\sum_{\ell \in \mathcal{Y}_j} \widetilde{L}_j^{(\ell)}} , \qquad (3)$$

where it is understood that $i \notin \mathcal{Y}_j$, i.e., i is not an upstream neighbor of j. Note that this is a way of "fairly" allotting more resources to nodes with fewer available routes, not a means to split the capacity of node j's links towards its upstream neighbors, which is instead taken care of distributedly via constraint (1c). Instead, we allow nodes with a single forwarding opportunity to convey all traffic there, while nodes with additional opportunities should split their traffic through all available routes.[3] For example, in Fig. 1, node 1 can only forward to node 5, so $F_5(1) = 1$, and because of constraint (1d), all of node 1's traffic will be conveyed through the link 1→5, which can transport $C_{5,\mathrm{LF}}^{(u)}$ bits over a time span u. Conversely, node 4 shares node 5 as a potential relay, but has an additional opportunity to forward to node 2: for this reason, $F_5(4) = 1/3$ and $F_2(4) = 2/3$, hence node 4 will send up to $C_{4,\mathrm{LF}}^{(u)} F_5(4)$ to node 5 and $C_{4,\mathrm{LF}}^{(u)} F_2(4)$ to node 2, as per constraint (1d).

$L_j^{(i)}$ in (2) is computed differently depending on the network topology information available to node i. We hereby distinguish between two cases: a) full topology-informed fair share computation, in case perfect topology information is available to i; and b) one-hop topology-informed fair share computation, otherwise.

In the case of the benchmark OMR–FF, where the nodes are aware of the full network graph, each node exactly computes the number of disjoint routes available to itself and to its neighbors. Clearly, this is not possible in the distributed case of our OMR algorithm. Here, only one-hop topology information is available to each node i. In this condition, every one-hop neighbor is a potential relay on a disjoint route to the destination. However, if some relays have common relays upstream, the number of disjoint routes available to i would be overestimated: in turn, i would be allotted a lower share of j's resources.

In order to minimize the route overestimation problem in OMR, we consider the nodes that are upstream neighbors of i (in set $\mathcal{Y}_i$) and neighbors of its considered upstream relay j (in set $\widetilde{\mathcal{Y}}_j$). We then eliminate at least the cases where a node $w \in \mathcal{Y}_i \cap \widetilde{\mathcal{Y}}_j$ has either i, j, or both as its only upstream neighbors. In fact, none of these cases would lead to a disjoint route. Formally, $L_j^{(i)}$ is estimated as

$$L_j^{(i)} = |\mathcal{Y}_i| - \sum_{w \in \mathcal{Y}_i \cap \widetilde{\mathcal{Y}}_j} \mathbb{1}\left[\mathcal{Y}_w \subseteq \{i, j\}\right] \qquad (4)$$

where $\mathbb{1}[\mathsf{p}]$ evaluates to 1 whenever the predicate p is true. We note that the computation of $L_j^{(i)}$ in (4) is not carried out if the destination $D \in \mathcal{Y}_j$. In this case, the traffic is always directed to the sink, without passing through other 1-hop neighbors.

---

[3]We remark that fair splitting is mainly needed in high traffic conditions. If the network traffic is sufficiently low, no bottlenecks will appear and the allocation will still be efficient.

Our fair resource splitting eliminates bottlenecks at the expense of favoring nodes with fewer routes to the sink by setting their fairness index, $F_j(i)$, to a high value. This fair splitting automatically adjusts to the network topology to optimize the channel utilization. Yet, due to this splitting, the performance may decrease in networks with low traffic, where allocation based only on link capacity is preferred. Moreover, in large networks, topology mismatches may lead to a wrong setting of the fairness index and thus to a performance degradation. Hence, our protocol mostly fits small underwater networks with a few hops to the sink (a scenario that occurs in the majority of underwater network applications), and those cases where either the network traffic is moderate to high, or when the traffic is unknown.

2) Calculation of upstream resources:
We proceed with the computation of Δ_j^τ from (1c), which represents the amount of j's upstream transmission resources that can be assigned to node i. This computation is made based on an estimate (obtained by i) of the quantities $\widehat{R}_j^\tau(k, t) \ \forall k \in \mathcal{Y}_j$. These quantities estimate the outcome of the allocation problem as may be solved by j to compute how many bits it should transmit to its own upstream neighbors with each technology $t \in \mathcal{T}_{j,k}^\tau$. We have

$$\Delta_j^\tau = \sum_{k \in \mathcal{Y}_j} \sum_{t \in \mathcal{T}_{j,k}^\tau} \left(C_{j,t}^{(u)} - \widehat{R}_j^\tau(k, t) \right). \qquad (5)$$

Note that we still indicate the current time τ as a reminder that the current solution to i's problem depends on j's solution for its current transmission allocation. The quantities $\widehat{R}_j^\tau(k, t)$ are obtained by i by solving the following problem:

$$\widehat{R}_j^\tau(k, t) = \arg\max_{R_j^\tau(k,t)} \sum_{k \in \mathcal{Y}_i} \sum_{t \in \mathcal{T}_{j,k}^\tau(k,t)} R_j^\tau(k, t) \qquad (6a)$$

$$\text{s.t.} \ \sum_{k \in \mathcal{Y}_j} \sum_{t \in \mathcal{T}_{j,k}^\tau} R_j^\tau(k, t) \leq P_{j,\tau}; \qquad (6b)$$

$$R_j^\tau(k, t) \leq C_{j,t}^{(u)} F_k(j), \qquad (6c)$$

where $F_k(j)$ is the share of node k's resources that can be devoted to transport node j's traffic. Constraint (6b) means that the bits transmitted through all technologies shall not exceed the remaining number of bits in node j's queue, whereas constraint (6c) implies that the number of bits transmitted by j via either technology shall not exceed its share of the upstream capacity of its relay k over a time period of length u. No constraint is imposed based on the terms Δ_k^τ, as i does not know them and it would take too many resources for j to transmit the corresponding information, especially over slow acoustic links. Note that (5) enforces congestion control in the network, by avoiding that a downstream node transmits more data than the receiving relay can advance towards D.

We note that the same procedure described in Section III-C1 above is employed to compute the fairness values $F_k(j)$ $\forall k \in \mathcal{Y}_j$. When full topology information is available, i.e., in the case of the benchmark OMR–FF, this procedure is trivial. However, since OMR is a distributed solution, node i cannot calculate $F_k(j)$ without knowledge of $\mathcal{Y}_k \ \forall k \in \mathcal{Y}_j$, and therefore must rely on node j to transmit the $F_k(j)$ values.

Similarly, since node i is not aware of $P_{j,\tau}$, we let j piggyback this value into each transmission. As a result, the overhead of this information is in the order of only a few bits. Assume that j communicated $P_{j,\tau'}$ at some preceding instant $\tau' < \tau$: $P_{j,\tau}$ can be readily derived as

$$P_{j,\tau} = P_{j,\tau'} - \sum_{k \in \mathcal{Y}_j} \sum_{t \in \mathcal{T}_{j,k}^{\tau'}} \widehat{R}_j^{\tau}(k,t). \qquad (7)$$

D. Complexity and overhead of OMR

To obtain the routing solution with OMR, each node i needs to solve (1) and (6). Since both $\widehat{R}_i^{\tau}(j,t)$ and $\widehat{R}_j^{\tau}(k,t)$ can take any value, these two optimization problems are solved through linear programming. The average complexity of OMR is therefore polynomial with $|\mathcal{Y}_i| \cdot \max_j |\mathcal{T}_{i,j}^{\tau}|$.

In terms of overhead, OMR requires the transmission of the size of the queue of one-hop neighbors, $P_{j,\tau}$. Representing $P_{j,\tau}$ as one byte, the total overhead of OMR is therefore $N^2 + N^2 \log(N) + 8N$ bits. OMR is fully distributed and requires information only from one-hop links to transfer $\mathcal{Y}_j$, $F_k(j)$, $\forall k \in \mathcal{Y}_j$, and $P_{j,\tau}$. This information is piggybacked to node i within the packet transmitted by its one-hop neighbor j only when the values change. Since the network topology changes very slowly, $\mathcal{Y}_j$ and $F_k(j)$ are rarely transmitted, and only $P_{j,\tau}$ must be updated after each packet transmission. The communication overhead of OMR is therefore only $8N$ bits.

E. Implementation details

OMR operates at the network layer. For node i, OMR receives original packets from the application layer, and packets to relay from $|\mathcal{T}_i|$ multimodal MAC layers, one for each communication technology. The packets are saved in a *routing buffer*, implemented as a FIFO queue, which advances based on notifications received from the multimodal MAC layers. Once these layers notify node i about the possibility to transmit, OMR calculates (1), and reads $\widehat{R}_i^{\tau}(j,t)$ bytes from its queue. To optimize channel use, we allow byte allocation from part of various awaiting packets. If $\widehat{R}_i^{\tau}(j,t)$ is smaller than the size of the next packet, then the packet is segmented. Otherwise, if $\widehat{R}_i^{\tau}(j,t)$ is larger than the size of the next packet, bytes belonging to multiple packets are transmitted. In both cases, the packets are rebuilt at the sink based on an identification number added as a header. Fig. 2 shows a typical packet format. Note that, to support OMR decisions in the neighborhood of a transmitting node, the header includes the number of remaining bytes in the sender's queue.

To solve problems (1) and (6), we use the simplex algorithm [40]. Both (1) and the techniques to obtain its parameters (for example, (3) and (6)) are formalized as linear optimization problems with convex lower bound constraints, and the solution found is globally optimal. The average complexity is polynomial [41], and the solution is thus scalable. For reproducibility, we publish the implementation of OMR.[4]

---

[4]The code is available for download at http://marsci.haifa.ac.il/share/diamant/MultiModalRoutingCode.zip.

Source ID	Packet ID (1)	Relay ID (1)	Start bit (1)	Length (1)	Payload (1)
Byte in queue	Packet ID (2)	Relay ID (2)	Start bit (2)	Length (2)	Payload (2)
⋮	Packet ID (n)	Relay ID (n)	Start bit (n)	Length (n)	Payload (n)

Fig. 2. Format of a typical transmitted datagram formed by fragments taken from n packets.

IV. SIMULATIONS

We now evaluate the performance of our OMR routing scheme through numerical simulations.

A. Benchmark Methods

We compare the performance of OMR with the following benchmarks.

Flooding — A node broadcasts all incoming packets through all available technologies. The flooding method will generate multiple copies of each packet. The result is high link utilization and, in the absence of packet collisions, small end-to-end delay as well as high goodput. As a result, flooding is a preferred choice in contention-free environments where the highest reliability is sought. However, the performance of this scheme is expected to decrease considerably when a contention-based MAC is used. Moreover, due to the many transmissions, energy efficiency is expected to be low.

RND — A node allocates the bits of an incoming packet uniformly at random across the available outgoing links. Like OMR, the RND scheme fragments packets for simultaneous transmission through different links. The main strength of the method is that it is simple and fully distributed with no need for any topology information. However, the allocation does not consider the flow constraints of upstream nodes.

OMR–FF — A version of OMR where the fair share of the transmission resources to be allotted to each node is calculated based on full topology information. OMR–FF stands in contrast to OMR, where the fair share computation is based only on the knowledge of one-hop links. This difference affects the way resources are allocated to different nodes over a multi-hop path: with only local topology information, OMR is more conservative in terms of link capacity usage than OMR–FF.

ETT — A packet is sent through a chosen link following the procedure in [31], modified to manage the case of multiple technologies available per link. Specifically, instead of the expected transmission time (ETT) metric $\text{ETT}_{k,i}$ for the kth path and the ith link along path k, we employ $\text{ETT}_{k,i,t}$ for technology t over the ith link. Therefore, we modified [31] as follows. To choose the path, we compute the weighted cumulative ETT (WCETT) metric as

$$\text{WCETT} = (1-\beta) \sum_i \left(\min_t \left[\text{ETT}_{k,i,t} \right] \right) + \beta \max(X_k), \quad (8)$$

where X_k is the accumulated delay over path k. Once a path k to the destination is chosen based on its WCETT metric, the technology t whose $\text{ETT}_{k,i,t}$ is maximum is chosen to

transmit over the ith link. Different than our distributed OMR solution but similar to the centralized benchmark OMR–FF, the ETT scheme requires global topology information. The ETT scheme finds the best route in terms of delivery time but neglects bottlenecks created due to the extensive use of such best routes by all nodes. The result is a less fair routing scheme, whose delivery time is expected to be high for high traffic, and whose performance is expected to deteriorate in the presence of packet collisions. Moreover, the strategy of ETT leads to high topology dependence, as bottlenecks are mostly expected in sparse networks.

To avoid loops in the flooding and RND schemes, we include in each packet the routing path it has traveled. With flooding, a receiver will avoid broadcasting a packet if the routing path shows that the packet has already traveled through all of the node's one-hop neighbors. In the RND scheme, no bit will be allocated to a given link if the packet has already traveled through the other endpoint of such link. In all benchmark methods, packets are fragmented according to the maximum length allowed by the technology through which the packet is sent. Last, the performance of the ETT method depends on the parameter $0 \leq \beta \leq 1$, which trades off accumulated delay vs. maximum delay. In the simulation results below, we explored the performance for various values of β, and show the results for the β value that leads to the best performance for each of the metrics introduced in the following subsection.

B. Quality metrics

We measure the performance in terms of end-to-end transmission delay, per-node goodput, message success rate, transmission efficiency, and link throughput. Once all fragments of a packet i of node n have been successfully received by the sink, we measure the message's end-to-end transmission delay as

$$\rho_d = \frac{1}{N-1} \sum_{n=1}^{N-1} \frac{1}{R_n} \sum_{i=1}^{R_n} (T_{n,i}^r - T_{n,i}^s) , \qquad (9)$$

where $T_{n,i}^r$ is the time when the full message was received, $T_{n,i}^s$ is the time when the message reached the network layer for routing, and R_n is the number of messages sent by node n and received in full by the sink node. For a network run time T_{net}, the per-node goodput is defined by

$$\rho_g = \frac{1}{N-1} \sum_{n=1}^{N-1} \sum_{i=1}^{I_n} \frac{M_{n,i}^r}{T_{\mathrm{net}}} , \qquad (10)$$

where $M_{n,i}^r$ is the number of bytes received by the sink for a message i originated from node n, and I_n is the number of messages originated by node n. The average per-node message success rate is

$$\rho_s = \frac{1}{N-1} \sum_{n=1}^{N-1} \frac{R_n}{I_n} . \qquad (11)$$

Note that $M_{n,i}^r$ from (10) can exceed the number of bytes transmitted by node n, denoted by $M_{n,i}^s$. This case happens when message i or parts of it are sent through several links such that the sink may receive multiple copies of some message chunks. We consider these cases as a waste of resources,

and call overhead the fraction of messages for which extra copies of any parts of the message are received by the sink. Formally:

$$\rho_o = \frac{1}{N-1} \sum_{n=1}^{N-1} \sum_{i=1}^{I_n} U \left(\frac{M_{n,i}^r}{M_{n,i}^s} - 1 \right) , \qquad (12)$$

where $U(x)$ is a step function whose value equals 1 if $x > 0$, and zero otherwise. We also consider an energy efficiency metric: this is defined by

$$\rho_e^{\mathrm{tot}} = \sum_{n=1}^{N-1} \left(\sum_{i=1}^{I_n} B_{i,n} T_{\mathrm{tx}}^{\mathrm{byte}} W_{\mathrm{tx}} \right) + T_{\mathrm{idle}}(n) W_{\mathrm{idle}} , \qquad (13)$$

where $B_{i,n}$ is the total number of bytes transmitted for message i originated from node n, $T_{\mathrm{tx}}^{\mathrm{byte}}$ and $T_{\mathrm{idle}}(n)$ are respectively the time spent in the transmission and idle state by node n, whereas W_{tx} and W_{idle} represent the power consumption in either state. As in this study we will mainly focus on multimodal acoustic networks (where a node integrates different acoustic PHYs), the transmit power outweighs the idle listening power. Considering also that W_{tx}, W_{idle} and $T_{\mathrm{tx}}^{\mathrm{byte}}$ are constants, we normalize our efficiency metric to the total number of transmitted bytes and to the total network operation time, and define it as the total number of transmitted bytes across the network for a single message:

$$\rho_e = \frac{1}{(N-1) \sum_{n=1}^{N-1} I_n} \sum_{n=1}^{N-1} \sum_{i=1}^{I_n} \frac{B_{i,n}}{T_{\mathrm{net}}} , \qquad (14)$$

Finally, the throughput of the link from node n to node m using communication technology t is defined as the ratio between the number of bytes successfully transmitted through the link, $R_{n,m}^t$, and the run time. Formally, the average link throughput is

$$\rho_u = \frac{1}{N^t} \sum_{n \in \mathcal{N}^t} \frac{1}{D_n^t} \sum_{m \in \mathcal{D}_n^t} \frac{R_{n,m}^t}{T_{\mathrm{net}}} , \qquad (15)$$

where $\mathcal{N}^t$ is the set of the nodes that hold communication technology t, and $\mathcal{D}_n^t$ is the set of the nodes that share a communication link with node n via technology t. Moreover, $|\mathcal{N}^t| = N^t$ and $|\mathcal{D}_n^t| = D_n^t$.

As mentioned in Section III-A, we aim at minimizing ρ_d, and at maximizing ρ_g and ρ_s. Yet, for energy conservation, we are also interested in minimizing ρ_o and ρ_e. Finally, for better fairness and to avoid congestion, we are interested in a large ρ_u.

C. Simulation setup

Our simulation setup is based on a Monte-Carlo set of 1000 network topologies. In each simulation run, $N = 10$ nodes are placed uniformly at random over an area of 500×500 m^2 with water depth of 100 m. The line of sight between the nodes may be interrupted by four horizontal obstacles and one vertical obstacle at uniformly distributed locations with uniformly distributed length in the range $[10, 50]$ m. Node 10 is defined as the sink node. Each of the other nine nodes is equipped with one or more communication technologies

TABLE II
SIMULATIONS: CHARACTERISTICS OF THE SIMULATED COMMUNICATION
TECHNOLOGIES

Technology	Bit rate [bps]	Max range [m]
Low-rate acoustics	1000	3000
Mid-rate acoustics	32000	300
High-rate acoustics	64000	100

at random between low-frequency, mid-frequency, and high-frequency acoustic communications. The characteristics of the three acoustic systems are based on the three EvoLogics modems working in the 18–34 kHz, 48–78 kHz, and 120–200 kHz bands [42]. A summary of these characteristics is provided in Table II, where the communication range of each model has been conservatively set. We remark that all acoustic technologies operate on mutually orthogonal bands, making the nodes multimodal. We run each simulation for $T_{net} = 600$ s. At the beginning of each simulation, each of the nine nodes generates its own packets according to a Poisson process of rate $\lambda = 3$ packets per minute per node. The size of each packet is drawn uniformly at random between 0 and 64 kbits. At any given time, the node is either idle, or serving a self-generated message or a packet received by another node. For each served packet, the node solves the routing allocation problem, as discussed in Section III. The packet is then segmented according to the solution of the routing problem and sent over the different links according to the determined routing allocation. Besides the information-bearing bytes, each packet segment includes the ID of the original message, the location of the packet segment within the original message, and the routing path the packet segment has gone through. Once received at the sink node, the various packet segments belonging to the same message are combined together.

We consider a binary phase-shift-keying modulation, and a scheduling protocol where a node holding a packet transmits it as soon as all its communication technologies are free. Once a packet is received, an acknowledgment is transmitted. To form the full topology information required for the OMR–FF method and the one-hop link information required for OMR, we refer to the communication ranges in Table II. For example, for mid-frequency acoustic communications, a link would be assumed to exist if the distance between the two nodes is smaller than 300 m, and this distance is continuously measured in our simulations by an underlying PHY mechanism. To calculate the route on the way to the sink (i.e., the sets $\mathcal{Y}_i, \forall i$), we carry out a preliminary route discovery phase, where the sink propagates a discovery packet through the network. The discovered routes are kept stable throughout each simulation run.

While in the OMR scheme the one-hop links are assumed to be known, in reality links would vary from the communication range set in Table II. To simulate this, we calculate the instantaneous packet error rate (PER) for each link used by transmitted packet segments. Once a packet transmission fails and/or no acknowledgment is received, the packet is shifted to the end of the message queue and is re-transmitted at a later time. The PER is computed based on the simulated signal-to-noise ratio (SNR) and on the packet size. The SNR of the low-rate and high-rate acoustic links is calculated using the Bellhop framework [43, Ch. 3] for shallow waters of depth 100 m, flat sandy bottom, fixed sound speed of 1500 m/s, and considering a source level of 170 dB re (1 μPa at 1 m). The ambient noise level is set to 40 dB re (1μPa^2/Hz) for low-frequency acoustics, as 30 dB re (1μPa^2/Hz) for mid-frequency acoustics, and 10 dB re (1μPa^2/Hz) for high-frequency acoustics.

We consider two MAC schemes. With the first, named *Ideal*, no packet collisions occur and acknowledgments are always received with no errors. This ideal protocol works in favor of the flooding scheme, where the links are expected to be utilized in full. The second (*Immediate*) is a MAC protocol in which packets are transmitted immediately upon arriving to the MAC layer, unless another transmission or reception is already taking place, and the reception of packets and acknowledgments is determined based on the link SNR and only when no collision occurs with another packet or acknowledgment. The Immediate MAC models the protocol [44] employed by the modems in the lake experiment (see Section V). In both the Ideal and Immediate MAC approaches, packets that need to be re-transmitted are re-inserted as new packets at the end of the queue.

D. Simulation results

In Figs. 3a and 3b, we show the cumulative distribution function (CDF) of the delay ρ_d for the Ideal MAC and the Immediate (realistic) MAC, respectively, computed over the ensemble of results from the whole Monte-Carlo simulation set. The CDF enables us to evaluate the entire distribution of the results. For the RND and ETT benchmarks, we observe a very wide span of results. This span suggests that these methods lack robustness, or in other words that thay do not provide sufficient performance guarantees over different topologies. By way of contrast, the distributions of the end-to-end delay of OMR and of its ideal benchmark OMR-FF (which counts on full network topology knowledge), span a much narrower set of values. We therefore conclude that these methods are robust to the network setup. The fact that the results span more than 100 s of delay is due to low-capacity links requiring packets to be segmented into small fragments. This tends to increase the backlog of the nodes, which in turn increases the delay. From Fig. 3a, we observe that with Ideal MAC the delay of the flooding scheme is better than that of OMR–FF and OMR and so is some of the performance of ETT. This result is obtained since in Ideal MAC we neglect multiple-access interference, hence flooding and ETT can propagate messages very fast through the network. Yet, the advantage of ETT is achieved in only 40% of the cases with an average of roughly 110 s compared to 39 s and 38 s for OMR–FF and OMR, respectively. This is because ETT obtains good results mostly in dense topologies. In terms of end-to-end delay, the performance of RND is the worst, since it does not optimize the packet allocation and generates bottlenecks with high probability. When the realistic Immediate MAC is used (Fig. 3b), we observe that all methods are affected.
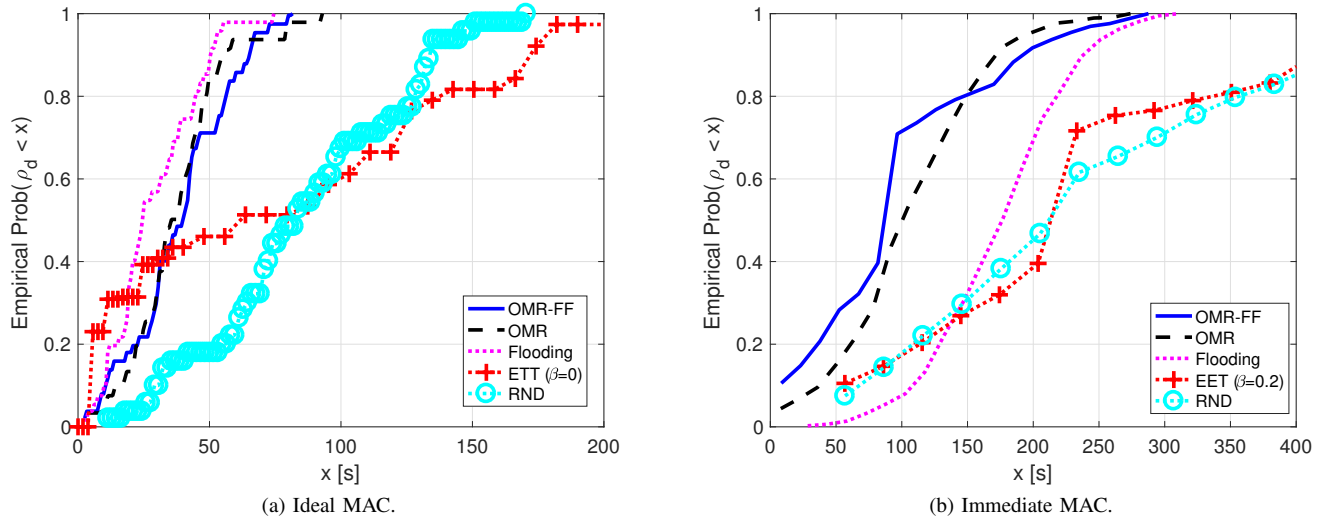
(a) Ideal MAC.

(b) Immediate MAC.

Fig. 3. Simulations: CDF of end-to-end transmission delay, ρ_d, from (9). Flooding achieves the best results with the Ideal MAC, which neglects collisions. With the more realistic Immediate MAC, OMR achieves the best results.
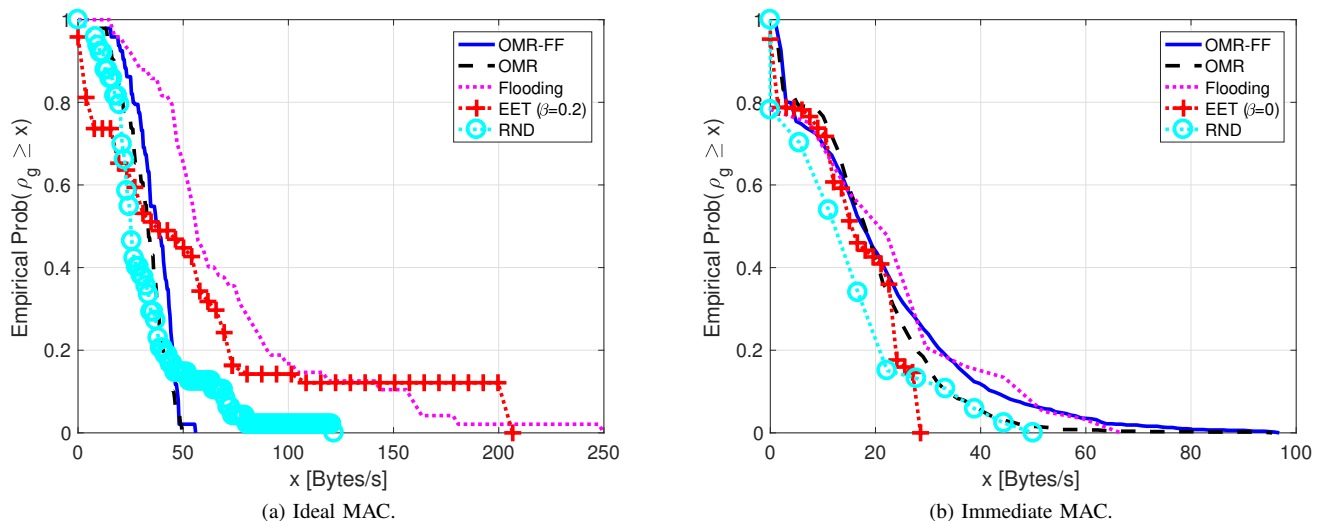

(a) Ideal MAC.

(b) Immediate MAC.

Fig. 4. Simulations: C-CDF of per-node goodput, ρ_g, from (10). The results show that OMR achieves similar performance as OMR–FF, and that when collisions are considered, the goodput of OMR is better than ETT and similar to flooding.

Yet, as noted above, the performance span of RND and ETT shows that these methods are less robust to varying topologies than the flooding scheme and our OMR method. As expected, the delay of OMR–FF is better than the delay of OMR for the realistic MAC case. This is because the availability of topology information in OMR–FF makes it possible for the nodes to optimally allocate transmission resources. However, we remark that the delay of our distributed OMR solution is almost as good as that of the benchmark centralized OMR-FF solution, which is remarkable given the limited topology information.

Next, in Figs. 4a and 4b, we show the complementary CDF (C-CDF) of the goodput, ρ_g, for the Ideal MAC and the Immediate MAC, respectively. The random bit allocation of RND yields low goodput values, since it does not give

priority to less stable links. Yet, with Ideal MAC, ETT obtains high goodput that is comparable to the energy-wasting flooding scheme. However, the performance of both ETT and flooding is more dispersed compared to OMR (meaning that OMR's performance is more predictable). Moreover, while the performance of ETT and flooding decreases much when packet collisions are considered (Fig. 4b), our OMR scheme achieves almost the same performance with either MAC, implying a good level of robustness. The results confirm that flooding and ETT outperform OMR only when packet collisions are ignored (Fig. 4a), otherwise the performance of ETT is worse than OMR's, and flooding achieves results similar to OMR (Fig. 4b). The reason for the former is the bottlenecks created by ETT, whereas the latter is due to the large number of packet collisions caused by the many transmissions of flooding. This
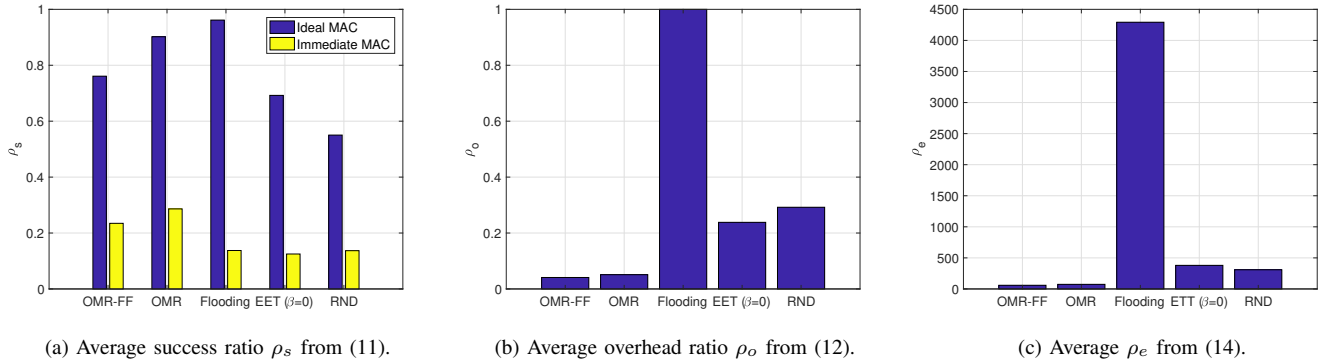
(a) Average success ratio ρ_s from (11).

(b) Average overhead ratio ρ_o from (12).

(c) Average ρ_e from (14).

Fig. 5. Simulations. Success rate and energy efficiency for the Immediate MAC protocol. OMR is more efficient than flooding, ETT, and RND. OMR–FF and OMR perform similarly.

is confirmed by the results of the lake experiment given further below.

When comparing the goodput of the five schemes, it is also of interest to examine the packet delivery ratio, ρ_s from (11), as shown in Fig. 5a for the Ideal and the Immediate MACs, respectively. Since the link utilization of RND is low, OMR outperforms RND. When the Ideal MAC is considered, the success rate of the flooding scheme is the highest, while the performance of ETT is slightly less than OMR's. In fact, flooding transmits each packet many times through different links, thus increasing the probability of correct delivery, whereas ETT creates several bottlenecks along the best path. However, for the immediate MAC, we observe that packet collisions reduce the delivery ratio of flooding (despite the high transmission redundancy of the scheme) and of ETT considerably. In fact, the increased load imposed on the queues of the nodes by the redundancy of flooding and the single chosen path of ETT actually contributes to the poor delivery ratios of these two schemes.

To comment on the energy efficiency of the three methods, in Figs. 5b and 5c we show the overhead ρ_o, and the total number of transmitted bytes, ρ_e, respectively, for the case of Immediate MAC. While multiple (redundant) copies of all messages are received with the flooding scheme, in the two OMR versions the sink receives extra copies only for about 8% of the messages. We also observe that although ETT sends packets along a single path, its overhead exceeds that of OMR. This is because the load on the path chosen by ETT tends to create many collisions also on the feedback channel, thereby leading to the retransmission of many unnecessary packets. Similarly, while RND does not issue redundant packets, its sub-optimal link utilization leads to the transmission of many more packets than required, and thereby to an increase of the number of collisions affecting acknowledgment packets. The advantage of OMR in terms of overhead is further emphasized by the huge difference in the energy efficiency measured as the total number of bytes sent by the three benchmark methods compared to OMR, and shown in Fig. 5c. Also here, this result slightly favors OMR, which requires much less knowledge and is thus more distributed.

To summarize, the simulation results show that OMR can deliver packets effectively to their destination, and at the same time it is an energy efficient method, that does not originate a large amount of overhead and that keeps the number of transmitted bits limited. Compared to the three benchmark schemes, the performance of OMR is less sensitive to the network topology. When a realistic MAC protocol is considered, where packet collisions can occur, OMR outperforms the three benchmark schemes in almost all categories. We also note that the above advantages are obtained without ETT's assumption that the packet error rate in the network are known. This is a hard assumption, that requires knowledge of the SNR at the receiver and likely of the transmission range in each link. Equally important, the advantage of OMR over ETT is obtained with only 1-hop topology information, compared to global topology information in ETT. Hence, OMR is also a more practical scheme than ETT.

V. FIELD EXPERIMENT

Our simulations revealed that the results of the comparison between OMR and the other benchmark schemes depend on whether packet collisions are considered or not. To support and complement these conclusions, we carried out an experiment in a real underwater environment. Experimenting with real systems includes non-ideal modem hardware behaviors, multipath propagation, actual packet collisions, the impact of finite memory in each node, and delays due to the management of multimodal technologies. In the following, we describe the setup of our field trial and the results obtained.

A. Setup of the experiment

The trial took place in June 2016, in the Werbellin lake, north of Berlin, Germany. The lake is narrow and long, with a maximum depth of 55 m. The resulting acoustic channel is characterized by a long delay spread and location-dependent ambient noise, which poses a significant challenge for underwater acoustic networks. The experiment included six nodes deployed at four different geographical locations. Three locations were reached using small vessels: two motorized inflatable boats, and one motorboat. The fourth location was one of the lake's docks. Throughout the experiment, the boats tended to drift at an approximate speed of 0.25 m/s.
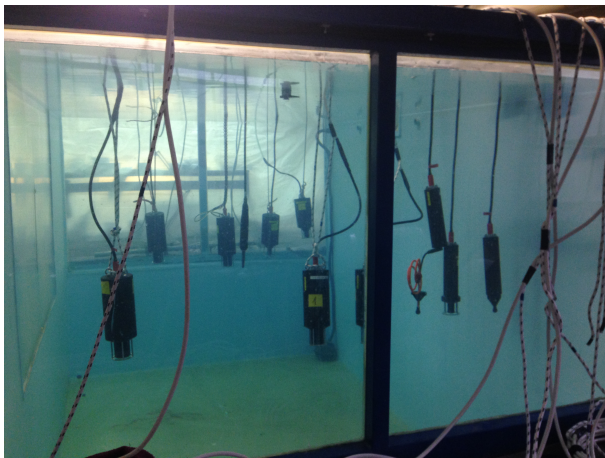
Fig. 6. Experiment: A picture taken in a water tank showing the ten underwater acoustic modems during preliminary system tests.

TABLE III
TECHNOLOGIES AVAILABLE TO EACH NODE IN EACH TOPOLOGY AND APPROXIMATE DEPLOYMENT DEPTH (BETWEEN PARENTHESES)

	Topology 1	Topology 2	Topology 3	Topology 4	Topology 5
Node 1	MF, HF (3 m)	MF, HF (3 m)	LF, MF (3 m)	MF (3 m)	MF (3 m)
Node 2	LF, HF (10 m)	LF, MF (10 m)	LF, MF, HF (10 m)	LF, MF, HF (10 m)	LF, MF, HF (10 m)
Node 3	LF, MF (10 m)	LF, HF (10 m)	LF, MF (10 m)	LF, MF (10 m)	LF, MF (10 m)
Node 4	LF (10 m)	LF, MF (10 m)	HF (10 m)	HF (10 m)	HF (10 m)
Node 5	LF (5 m)	LF (10 m)	LF (5 m)	LF (10 m)	LF (10 m)
Node 6	LF (10 m)	LF (10 m)	LF (10 m)	LF (10 m)	LF (5 m)

The multimodal functionality was obtained via three types of EvoLogics acoustic modems [42], These integrated underwater communication systems include an acoustic transducer, driving electronics, a DSP/FPGA implementing the S2C modulation scheme [45], and a communications stack running on an ARM processor. The S2C scheme modulates a linear frequency-swept carrier signal through phase keying. The receiver is thus able to decouple delayed multipath arrivals in the frequency domain, reducing or eliminating frequency-selective fading.

A low-rate, low-frequency (*LF*) technology was incorporated by the S2C 18-34 modem, having a maximum transmission range of 3.5 km. A mid-frequency (*MF*) technology was obtained via the S2C 48-78 modem, which has a maximum range of 1 km. Finally, high-frequency (*HF*) communications were represented by the S2CM HS model, which is employed over short links of up to 300 m.

All modems were configured to transmit in instant message (IM) mode: this enacts the Immediate MAC protocol discussed in Section IV. Moreover, it allows us to customize both the header and the payload of all packets. We interfaced the modems to our software through the Matlab-to-Modem extension of DESERT Underwater [46], which is composed of a background C++ process and a foreground Matlab instance. For each technology, the C++ process manages an output buffer that contains data for the modem to transmit, and an input buffer that contains the payload of the instant messages received from the modem. Transmission and reception from each modem are performed via the EvoLogics drivers of DESERT Underwater [25]. We configure the native acknowledge (ACK) transmission feature of the modem IMs, allowing up to one retransmission. If this procedure fails, we re-insert the data in the queue for a later attempt. The Matlab process generates traffic, runs the routing protocol, and issues transmissions by writing output buffers and triggering the C++ process. It also processes input buffers and manages the received data according to the OMR rules. System pre-tests in a tank using the minimum source level allowed (see Fig. 6) revealed that the modems could work in parallel without generating any significant outband interference. Therefore, a proof-of-concept demonstration of our multimodal routing scheme is feasible using these modems. In total, we employed ten systems: five LF, three MF, and two HF modems. We remark that the HF acoustic technology closely resembles optical communications, in that HF modems provide a higher bit rate than common acoustic systems. Moreover, they can only be operated at close distances, such as encountered during AUV docking operations, or during close interactions among divers.

The performance of the routing schemes was tested in five different network topologies, illustrated in Fig. 7, where solid lines represent a communication link, and we mark the LF, MF, and/or HF communication technologies available to each node. In order to deploy the nodes, the boats moved to several waypoints in the lake as shown in Fig. 7. The figure also shows the location of each of the four stations, the respective distances, and the ID of the nodes in each station. As shown in the figure, node 1 was always deployed on the pier, nodes 5 and 6 were hosted in one inflatable boat each, whereas nodes 2, 3 and 4 were deployed from opposite ends of the motorboat.

In all topologies, node 6 served as the sink node. The modems were deployed at roughly one half of the local water column depth. Table III shows the mapping between the nodes and the available technologies in each scenario, along with the approximate deployment depth. Most of the required reconfigurations involve the shorter-range technologies MF and HF available to nodes 2, 3 and 4.

The topologies were designed to offer different routing scenarios by varying connectivity options over different technologies. This diversifies the routing opportunities available to each node. For example, in Topology 2, nodes 2, 3 and 4 have two technologies each. The resulting network configuration allows node 4 to forward packets across the following four different routes: a) $4 \xrightarrow{\text{LF/MF}} 2 \xrightarrow{\text{MF}} 6$; b) $4 \xrightarrow{\text{LF}} 5 \xrightarrow{\text{LF}} 2 \xrightarrow{\text{MF}} 6$; c) $4 \xrightarrow{\text{LF}} 5 \xrightarrow{\text{LF}} 3 \xrightarrow{\text{HF}} 6$; and d) $4 \xrightarrow{\text{LF/MF}} 2 \xrightarrow{\text{LF}} 5 \xrightarrow{\text{LF}} 3 \xrightarrow{\text{HF}} 6$, where above each arrow we indicate the technology used over the corresponding link. The availability of different opportunities for different nodes can be easily verified for each topology. Each node was driven by a laptop which ran the routing logic and drove the modems. Note that routing was performed in a distributed fashion.

For each of the five topologies, we conducted three experiments of 10 minutes each, one employing OMR, one with the

(a) Topology 1   (b) Topology 2
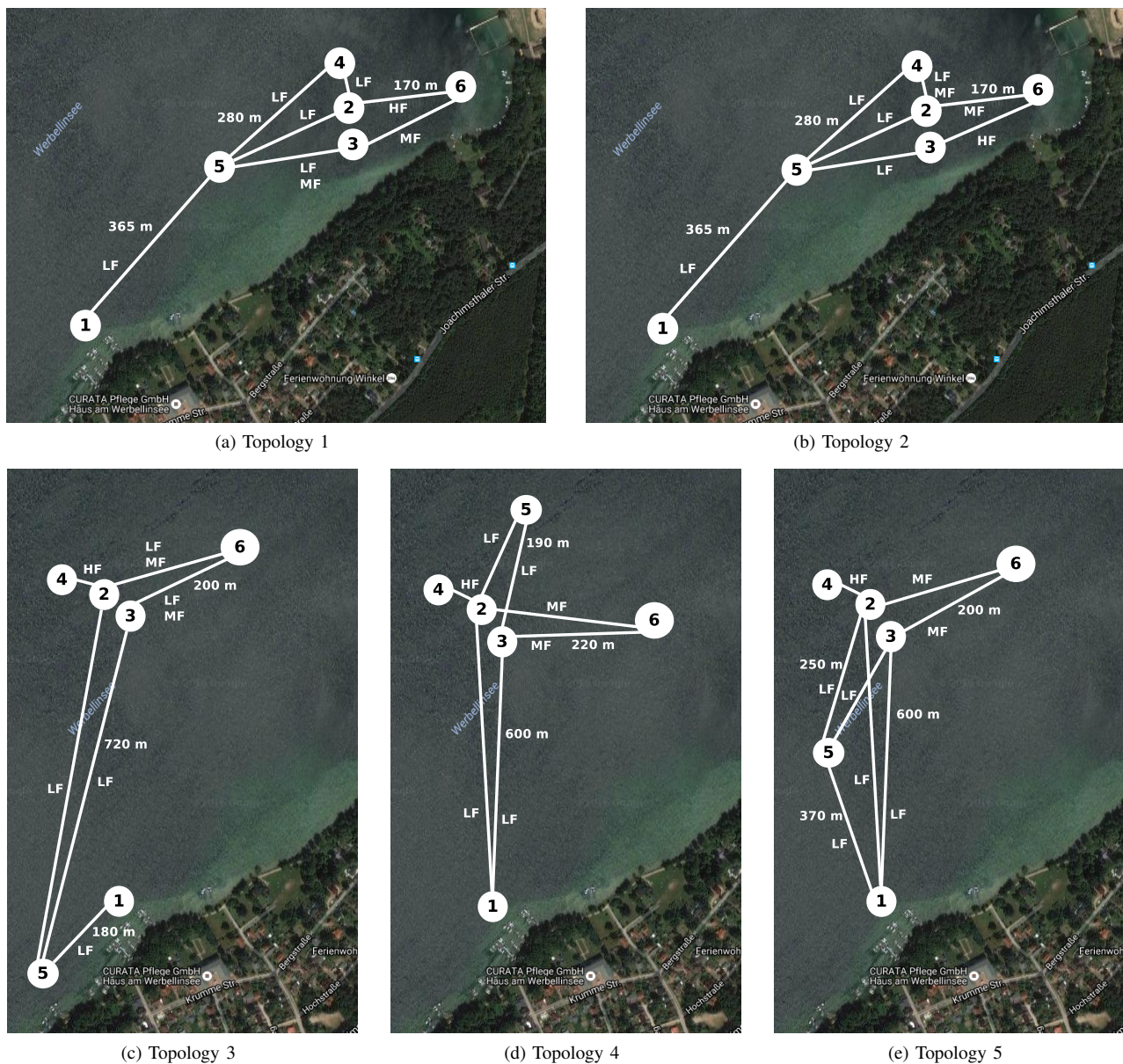
(c) Topology 3   (d) Topology 4   (e) Topology 5

Fig. 7. Logical network topology configurations and locations of the nodes in the five scenarios considered in our lake experiment. Each link is tagged with the technologies that can be used over that link.

centralized benchmark OMR–FF, and one with the benchmark flooding scheme described in Section IV-A. The information regarding the communication technologies available in each topology (one-hop links for OMR, or full topology information for OMR–FF) was obtained via a preliminary link discovery phase [37].[5]

Implementing the five experiment topologies required the coordination of the three boats and the pier in order to establish the required links and at the same time avoid unwanted interference. While logical topology configurations can be practically achieved by grouping the nodes close to one another, and using software filters to remove unwanted links [38], this solution would not remove unwanted interference. For this reason, we preferred to move the boats to suitable locations

and check the topology discovered through the method in [37] until the desired topology was achieved. The environment was leveraged to facilitate the process whenever possible. For example, we hid unwanted links by lowering modems behind underwater slopes, or by moving boats inside harbors.

We lightly anchored the boats while carrying out each experiment. Upon the occurrence of significant drifts that would disrupt the desired topology, we repeated the corresponding experiment. We remark that meteorological conditions also have an impact on the obtained results. Strong wind and rain can inject unwanted noise in the water and constrain the nodes to remain physically closer in order to establish a link. To reduce the impact of these conditions on our results, we repeated all experiments for all topologies over two subsequent days. Fig. 8 shows two pictures taken during the experiment: on the left, a connectivity check is being performed between

---

[5]As we assume a MAC protocol that handles both link discovery and transmissions, neither adds to the routing overhead.

the boat with nodes 2, 3 and 4 to verify the communication of the modems in the water; on the right, nodes 2, 3 and 4 are controlled from a semi-rugged laptop computer.

In each experiment, each node generated its own set of data packets according to a Poisson process of rate $\lambda = 2$ packets per minute per node. This set was equal throughout all experiments. The size of each packet was drawn uniformly at random between 0 and 64 kbits. During each experiment, the nodes sent the data packets through multiple hops towards the sink, abiding to the rules of the OMR protocol presented in Section III or the flooding scheme described in Section IV. Periodically, the nodes exchanged information related to the number of packets in their queue, their neighbor lists and the remaining information needed to run the protocol. When operating OMR, the reception of each data packet was separately acknowledged. In case an ACK was not received, the packet was retransmitted once by the modem's MAC protocol. Broadcast packets (e.g., reporting the queue status in the OMR protocol and the hop history in the flooding protocol) were not acknowledged.

B. Results

With five topologies tested, we measure the performance of the experiment in terms of the end-to-end transmission delay ρ_d in (9), the goodput ρ_g in (10), and the link throughput ρ_u in (15). The end-to-end delay of each message was calculated only once the sink (node 1) received the message in full, while the goodput was calculated for each message segment received by the sink. For the link throughput, we considered any successful transmission in the link regardless of whether the packet segment was ultimately received by the sink or not.

We initially focus on Topology 1 (see Fig. 7), and start by discussing a per-link transmission rate sample in Fig. 9. We observe that flooding (Fig 9a) is too aggressive in transmitting packets over all available links, and results in poor rates (e.g., over the link from node 4 to node 5) or unnecessarily high rates (e.g., from node 5 to node 2). By cross-checking transmission and reception logs we noted that the main reason is that flooding is subject to a high chance of collisions, and to the

high bit error rate that results. On the contrary, the two OMR versions convey traffic more reliably through the network, resulting in an optimized utilization of the network link. In particular, OMR (that has no access to topology information beyond first-hop neighbors) tends to be more conservative (Fig 9b). As a consequence, the transmission rates of node 3's MF link and of the LF links of nodes 1, 4 and 5 are limited. Full topology awareness in OMR–FF makes nodes 1 and 4 aware of the capacity of node 5's upstream links, so that they can push more traffic through their LF links to node 5. In turn, node 5 will convey this to node 3 through both the MF and the LF links, and finally to node 6 through node 3's MF link. The overall result is higher transmission rates over all technologies (Fig 9c), and a lower amount of time required for a given number of packets to reach the sink (node 6) correctly.

In Fig. 10a, we show the measured end-to-end delay ρ_d (see (9)) for flooding and OMR in each topology. We observe that the end-to-end delay of flooding is significant, due to the many collisions (and subsequent retransmissions) caused by the forwarding of every packet over every available technology. While the centralized benchmark OMR-FF achieves better results than OMR, in some cases our distributed OMR achieved shorter transmission delay than OMR–FF. This is because OMR in general uses more links than OMR–FF, which tends to be an advantage in the presence of many collisions. The delay performance in Fig. 10a show that OMR achieves roughly the same results for topologies 1, 4, and 5, which are significantly better than the performance for topologies 2 and 3. Yet, the OMR-FF benchmark achieved about the same performance throughout the experiment. This is mostly because in the latter group of topologies there are many bottleneck nodes. Since OMR is a distributed scheme, it may not always find routes that circumvent bottlenecks. As a result, packets were delayed in bottlenecks before arriving to their destinations. Conversely, the global topology knowledge of the ideal benchmark OMR-FF makes it possible to find alternative routes.

Fig. 10b shows the goodput ρ_g from (10). We observe that due to the higher number of packet collisions in a real environment, the goodput of flooding decreased compared to
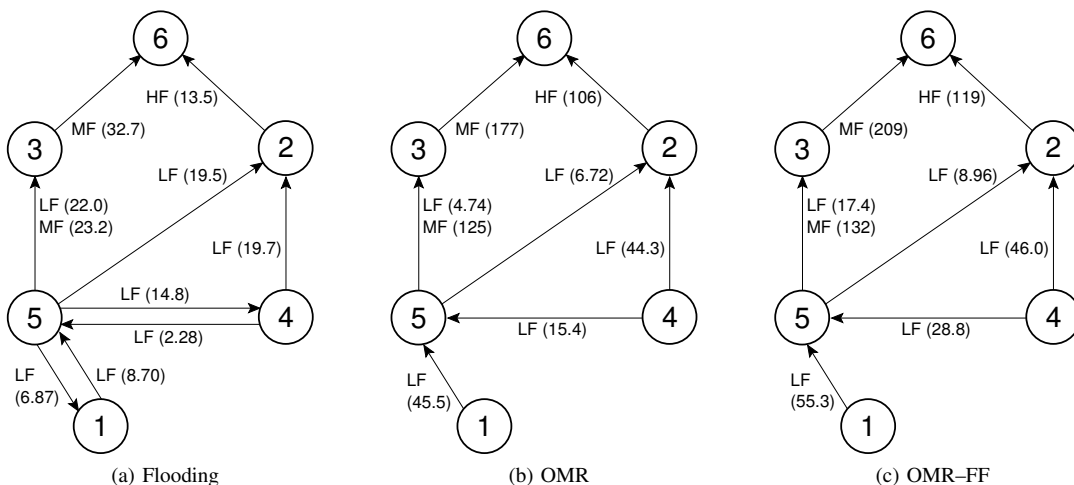

Fig. 9. Per-link transmission rates in bytes/s for all protocols run in the experiment, Topology 1.
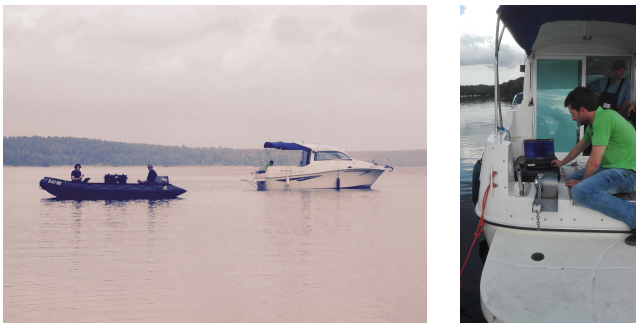
Fig. 8. Pictures taken during the lake experiment. (Left) Coordination between the boat containing nodes 2, 3 and 4, and the boat with node 6; (Right) Control of nodes 2, 3 and 4 through a semi-rugged laptop computer.

the simulations, becoming similar and sometimes lower than that of OMR–FF. Due to the use of full topology information, the goodput of the centralized OMR–FF is higher than that of our distributed OMR. An exception to the latter result is seen in Topology 2: the reason is that this topology offers many similar routes from each node to the sink, and thus spreading the transmissions over multiple links has a positive effect.

The per-topology link throughput ρ_u, (see (15)), is shown in Fig. 11 for each of the three communication technologies. In all cases, we observe that the centralized benchmark OMR–FF delivers the best performance and that, although flooding produces many more transmissions over each link, the link throughput of OMR is significantly higher. Again, this is a consequence of the many packet collisions that occur. For the same reason, the link throughput of the centralized benchmark OMR–FF is better than that of OMR. Comparing the link throughput for the three communication types, we observe that OMR channels more transmissions through links with higher capacity. As a result, the network adapts itself to the topology, as confirmed by the changes in the link throughput for the five topologies tested in the experiment, each having a different configuration of multimodal links.

VI. Conclusions

In this paper, we considered the network operation of multimodal underwater systems. We proposed OMR that, to the best of our knowledge, is the first optimal distributed routing protocol to be specifically designed for multimodal underwater networks and to be experimented in the field. Our protocol leverages local topology knowledge to decide how traffic should be distributed over available links using different communication technologies. This is achieved in a way that does not congest the relays upstream, and reserves more resources for the nodes with fewer routing opportunities. We analyzed the performance of OMR by means of both simulations and field experiments. Our results show that our protocol leverages the available technologies to deliver data reliably without congesting the network, even in the presence of limited, one-hop topology information. This suggests that the implementation of OMR, which is fully distributed, is a good solution to the routing problem in multimodal underwater networks.

While having additional communication technologies available provides clear advantages, it also provides a broader set of alternatives for a malicious user to try compromising the network or part of its nodes. For this reason, future work on this topic will specifically exploit multimodal network resources in order to enhance the security of such networks.

References

[1] R. Diamant, P. Casari, F. Campagnaro, and M. Zorzi, "Routing in multi-modal underwater networks: a throughput-optimal approach," in *Proc. WCNEE (IEEE INFOCOM Workshop)*, Atlanta, GA, May 2017.

[2] M. Chitre, S. Shahabudeen, and M. Stojanovic, "Underwater acoustic communications and networking: Recent advances and future challenges," *Marine Tech. Soc. Journal*, vol. 42, no. 1, pp. 103–116, Spring 2008.

[3] N. Farr, A. Bowen, J. Ware, C. Pontbriand, and M. Tivey, "An integrated, underwater optical/acoustic communications system," in *Proc. MTS/IEEE OCEANS*, Sydney, Australia, May 2010.

[4] C. Moriconi, G. Cupertino, S. Betti, and M. Tabacchiera, "Hybrid acoustic/optic communications in underwater swarms," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[5] "Sonardyne BlueComm Optical Modem," Accessed: Aug. 2017. [Online]. Available: http://www.sonardyne.com/products/all-products/instruments/1148-bluecomm-underwater-optical-modem.html

[6] X. Che, I. Wells, G. Dickers, P. Kear, and X. Gong, "Re-evaluation of RF electromagnetic communication in underwater sensor networks," *IEEE Commun. Mag.*, vol. 48, no. 12, pp. 143–151, Dec. 2010.

[7] "WFS Seatooth," accessed: Aug. 2017. [Online]. Available: http://www.wfs-tech.com/index.php/products/seatooth/

[8] T. Esemann, G. Ardelt, and H. Hellbrück, "Underwater electric field communication," in *Proc. ACM WUWNet*, Nov. 2014.

[9] N. Ahmed, J. Hoyt, A. Radchenko, D. Pommerenke, and Y. R. Zheng, "A multi-coil magneto-inductive transceiver for low-cost wireless sensor networks," in *Proc. UComms*, Sestri Levante, Italy, Sep. 2014.

[10] P. A. van Walree, "Propagation and scattering effects in underwater acoustic communication channels," *IEEE J. Ocean. Eng.*, vol. 38, no. 4, pp. 614–631, Oct. 2013.

[11] H. S. Dol, P. Casari, T. van der Zwan, and R. Otnes, "Software-defined underwater acoustic modems: Historical review and the NILUS approach," *IEEE J. Ocean. Eng.*, vol. 42, no. 3, pp. 722–737, Jul. 2017.

[12] F. Campagnaro, F. Favaro, P. Casari, and M. Zorzi, "On the feasibility of fully wireless remote control for underwater vehicles," in *Proc. Asilomar Conf. SS&C*, Pacific Grove, CA, Nov. 2014.

[13] H. Dol, M. Colin, P. van Walree, and R. Otnes, "How smartphone industry made UComms easier: Moore's law goes underwater," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[14] F. Campagnaro, F. Favaro, F. Guerra, V. Sanjuan, M. Zorzi, and P. Casari, "Simulation of multimodal optical and acoustic communications in underwater networks," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[15] I. Vasilescu, K. Kotay, D. Rus, P. Corke, and M. Dunbabin, "Data collection, storage and retrieval with an underwater optical and acoustical sensor network," in *Proc. ACM Sensys*, San Diego, CA, Nov. 2005.

[16] M. Stojanovic and L. Freitag, "Recent trends in underwater acoustic communications," *Marine Technology Society Journal*, vol. 47, no. 5, pp. 45–50, Sep. 2013.

[17] A. Tennenbaum, M. Dyakiw, J. H. Cui, and Z. Peng, "Application of low cost optical communication systems to underwater acoustic networks," in *Proc. IEEE MASS*, Philadelphia, PA, Oct. 2014.

[18] M. O'Rourke, E. Basha, and C. Detweiler, "Multi-modal communications in underwater sensor networks using depth adjustment," in *Proc. ACM WUWNet*, Los Angeles, CA, Nov. 2012.

[19] R. Kastner, A. Lin, C. Schurgers, J. Jaffe, P. Franks, and B. S. Stewart, "Sensor platforms for multimodal underwater monitoring," in *Proc. IGCC*, San Jose, CA, Jun. 2012.
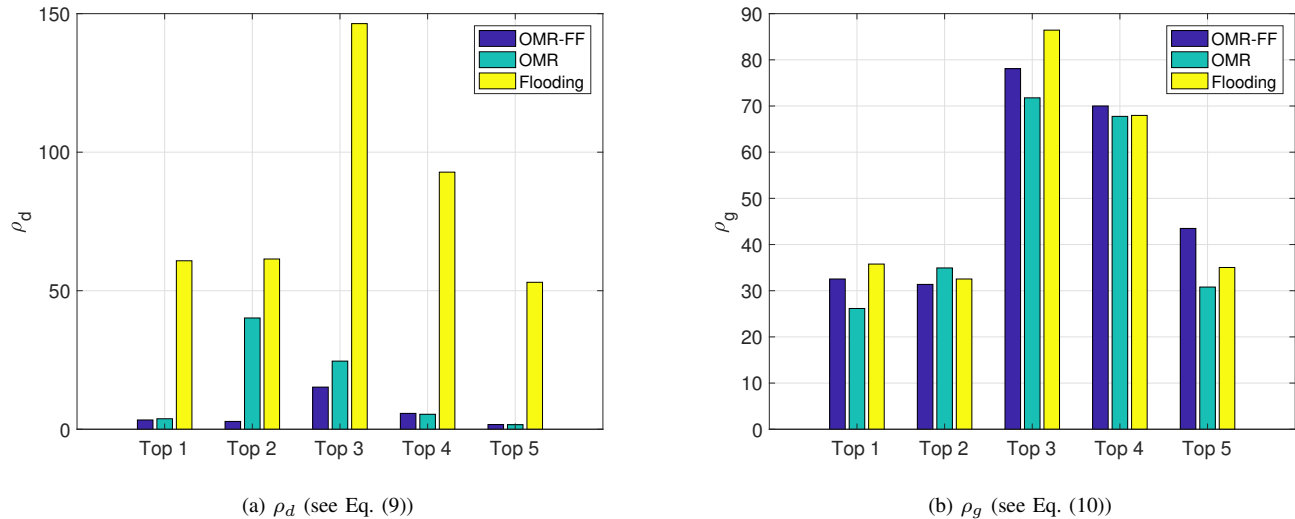
(a) ρ_d (see Eq. (9))

(b) ρ_g (see Eq. (10))

Fig. 10. Experiment: ρ_d and ρ_g. Flooding performs worse compared to the simulations. The centralized benchmark OMR–FF performs only slightly better than our distributed OMR.
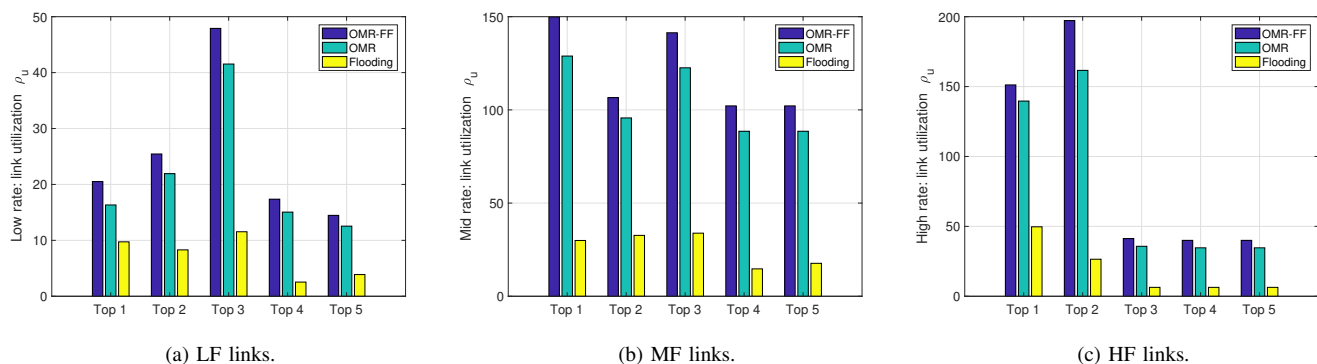

(a) LF links.

(b) MF links.

(c) HF links.

Fig. 11. Experiment: ρ_u. Flooding performs worse compared to the simulations. The centralized benchmark OMR–FF performs only slightly better than our distributed OMR.

[20] B. Benson, Y. Li, B. Faunce, K. Domond, D. Kimball, C. Schurgers, and R. Kastner, "Design of a low-cost underwater acoustic modem," *IEEE Embedded Syst. Lett.*, vol. 2, no. 3, pp. 58–61, Sep. 2010.

[21] R. Kastner, A. Lin, C. Schurgers, J. Jaffe, P. Franks, and B. S. Stewart, "MURAO: A multi-level routing protocol for acoustic-optical hybrid underwater wireless sensor networks," in *Proc. IEEE SECON*, Seoul, South Korea, Jun. 2012.

[22] S. Han, R. Chen, Y. Noh, and M. Gerla, "Real-time video streaming from mobile underwater sensors," in *Proc. ACM WUWNet*, Rome, Italy, Nov. 2014.

[23] S. Basagni, L. Böölöni, P. Gjanci, C. Petrioli, C. A. Phillips, and D. Turgut, "Maximizing the value of sensed information in underwater wireless sensor networks via an autonomous underwater vehicle," in *Proc. IEEE INFOCOM*, Toronto, Canada, Apr. 2014, pp. 988–996.

[24] F. Campagnaro, F. Guerra, R. Diamant, P. Casari, and M. Zorzi, "Implementation of a multimodal acoustic-optic underwater network protocol stack," in *Proc. MTS/IEEE OCEANS*, Shanghai, China, Apr. 2016.

[25] P. Casari *et al.*, "Open-source suites for underwater networking: WOSS and DESERT Underwater," *IEEE Network, special issue on "Open Source for Networking: Development and Experimentation"*, vol. 28, no. 5, pp. 38–46, Sep. 2014.

[26] R. Petroccia, G. Cario, M. Lupia, V. Djapic, and C. Petrioli, "First in-field experiments with a bilingual underwater acoustic modem supporting the JANUS standard," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[27] M. Radi, B. Dezfouli, K. A. Bakar, and M. Lee, "Multipath routing

in wireless sensor networks: Survey and research challenges," *MDPI Sensors*, vol. 12, no. 1, pp. 650–685, Jan. 2012. [Online]. Available: http://www.mdpi.com/1424-8220/12/1/650

[28] P. Kyasanur and N. H. Vaidya, "Routing and interface assignment in multi-channel multi-interface wireless networks," in *Proc. IEEE WCNC*, New Orleans, LA, Mar. 2005.

[29] M. X. Gong and S. F. Midkiff, "Distributed channel assignment protocols: a cross-layer approach," in *Proc. IEEE WCNC*, New Orleans, LA, Mar. 2005.

[30] A. Raniwala and T. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proc. IEEE INFOCOM*, Miami, FL, Mar. 2005.

[31] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. ACM MobiCom*, Philadelphia, PA, Sep. 2004.

[32] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. IEEE WMCSA*, New Orleans, LA, 1999.

[33] C. Pan, C. Liu, H. Zhou, and L. Gui, "Multi-path routing for video streaming in multi-radio multi-channel wireless mesh networks," in *Proc. IEEE ICC*, Kuala Lumpur, Malaysia, May 2016.

[34] I. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Elsevier Ad Hoc Networks*, vol. 3, no. 3, pp. 257–279, May 2005.

[35] Y. Sun and T. Melodia, "The internet underwater: An IP-compatible protocol stack for commercial undersea modems," in *Proc. ACM WUWNet*, Kaohsiung, Taiwan, Nov. 2013.

[36] A. Camillò, M. Nati, C. Petrioli, M. Rossi, and M. Zorzi, "IRIS:

Integrated data gathering and interest dissemination system for wireless sensor networks," *Elsevier Ad Hoc Networks*, vol. 11, no. 2, pp. 654–671, Mar. 2013.

[37] R. Diamant, R. Francescon, and M. Zorzi, "Efficient link discovery for underwater networks," in *Proc. UCOMMS*, Lerici, Italy, Aug. 2016.

[38] G. Toso, R. Masiero, P. Casari, O. Kebkal, M. Komar, and M. Zorzi, "Field experiments for dynamic source routing: S2C evologics modems run the SUN protocol using the DESERT Underwater libraries," in *Proc. MTS/IEEE OCEANS*, Hampton Roads, VA, Oct. 2012.

[39] S. Basagni, C. Petrioli, R. Petroccia, and D. Spaccini, "CARP: A channel-aware routing protocol for underwater acoustic wireless networks," *Elsevier Ad Hoc Networks*, vol. 34, pp. 92–104, Nov. 2015.

[40] J. Nelder and R. Mead, "A simplex method for function minimization," *The computer journal*, vol. 7, no. 4, 1965.

[41] A. Schrijver, *Theory of linear and integer programming*. John Wiley & Sons, 1998.

[42] "EvoLogics Underwater Acoustic Modems," last time accessed: Aug. 2017. [Online]. Available: https://www.evologics.de/en/products/acoustics/index.html

[43] F. Jensen, W. Kuperman, M. Porter, and H. Schmidt, *Computational Ocean Acoustics*, 2nd ed. Springer, 2000.

[44] O. Kebkal, M. Komar, and K. Kebkal, "D-MAC: Hybrid media access control for underwater acoustic sensor networks," in *Proc. IEEE ICC*, Cape Town, South Africa, May 2010.

[45] K. G. Kebkal and R. Bannasch, "Sweep-spread carrier for underwater communication over acoustic channels with strong multipath propagation," *The Journal of the Acoustical Society of America*, vol. 112, no. 5, pp. 2043–2052, 2002.

[46] F. Campagnaro, R. Francescon, F. Favaro, F. Guerra, R. Diamant, P. Casari, and M. Zorzi, "The DESERT Underwater framework v2: Improved capabilities and extension tools," in *Proc. UComms*, Lerici, Italy, 2016.

Paolo Casari (S'05–M'08–SM'13) received the PhD in Information Engineering in 2008 from the University of Padova, Italy. After being on leave at the Massachusetts Institute of Technology in 2007, his research progressively focused on underwater communications and networks. He collaborated to several funded projects including CLAM (FP7), RACUN (EDA), and several US ARO, ONR and NSF initiatives. He is the PI of the NATO SPS project ThreatDetect, and the scientific coordinator of the EU H2020 RECAP project. Previously, he was Technical Manager of the NAUTILUS and WISEWAI projects. In 2015, he joined the IMDEA Networks Institute, Madrid, Spain, where he leads the Ubiquitous Wireless Networks group. He served in the organizing committee of several conferences, and has been guest editor of a special issue of the Hindawi Journal of Electrical and Computer Engineering on "Underwater Communications and Networking," and is currently guest editor of a special issue of IEEE ACCESS on the same topic. His research interests include many aspects of underwater communications, such as channel modeling and simulation, network protocol design and evaluation, localization, and field experiments.
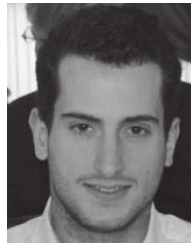
Filippo Campagnaro received the bachelor's degree in information engineering and the master's degree in telecommunication engineering from the University of Padova, Italy, in 2012 and 2014, respectively. He is currently pursuing the Ph.D. degree with the SIGNET Group, DEI. His master's thesis involved the design of a multimodal remote control for underwater remotely operated vehicles. He has participated in many sea trials with the NATO STO CMRE (La Spezia, Italy), EvoLogics GmbH (Berlin, Germany), and ENEA (Rome, Italy). He collaborates with the IMDEA Network Institute, Madrid, Spain, and NATO STO CMRE, to develop and maintain a simulator for underwater optical communications, based on real field measurements. He also collaborates with the University of Haifa, Israel, and Texas A&M, College Station, TX, USA, with the development of a module that will be integrated into the THEMO Marine Observatory which is off the coast of Israel. His current work involves software development for the DESERT Underwater framework, and the design of novel communication stacks architectures, with special attention to cross-layer and multi-technology design. His research interests revolve mainly around the design, analysis, implementation, and field evaluation of multimodal underwater networks.

Oleksiy Kebkal received his M.Sc. degree in Computer Sciences from the National Technical University of Ukraine (Kiev, Ukraine) in 2001, and is currently working towards his PhD degree. He held a research fellowship at the Technical University Berlin in 2002-2006 and joined EvoLogics GmbH in 2007. Mr. Kebkal is the company's leading expert in communication protocol design and signal processing, and developed the medium access algorithms of the EvoLogics D-MAC protocol.


Roee Diamant received his PhD from the Department of Electrical and Computer Engineering, University of British Columbia, in 2013, and his B.Sc. and the M.Sc. degrees from the Technion, Israel Institute of Technology, in 2002 and 2007, respectively. From 2001 to 2009, he worked in Rafael Advanced Defense Systems, Israel, as a project manager and systems engineer, where he developed a commercial underwater modem with network capabilities. In 2015 and 2016, he was a visiting Prof. at the University of Padova, Italy. In 2009, he received the Israel Excellent Worker First Place Award from the Israeli Presidential Institute. In 2010, he received the NSERC Vanier Canada Graduate Scholarship. Dr. Diamant has received three Best Paper awards, and serves as an associate editor for the IEEE Ocean Engineering. Currently, he is the coordinator of the EU H2020 project SYMBIOSIS (BG-14 track), and leads the underwater Acoustic and Navigation Laboratory (ANL) as an assist. Prof. at the Dept. of Marine Technology, University of Haifa. His research interests include underwater acoustic communication, underwater navigation, object identification, and classification.

Veronika Kebkal received the MS degree in Applied Mathematics from Moscow State University in 2009 with the Diploma Thesis "Algorithms and Software Development for turbo-coding of S2C-DQPSK signals in underwater acoustic communications." It was the beginning of her scientific activity in the field of underwater acoustic communications. Currently, she is working as software development engineer at Evologics GmbH, and focuses on the further development of the MAC and network layers for the acoustic modems built upon the Sweep-Spread Carrier (S2C) technology.

Michele Zorzi (S'89–M'95–SM'98–F'07) received his Laurea and PhD degrees in electrical engineering from the University of Padova in 1990 and 1994, respectively. During academic year 1992-1993 he was on leave at the University of California at San Diego (UCSD). In 1993 he joined the faculty of the Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy. After spending three years with the Center for Wireless Communications at UCSD, in 1998 he joined the School of Engineering of the University of Ferrara, Italy, where he became a professor in 2000. Since November 2003 he has been on the faculty of the Information Engineering Department at the University of Padova. His present research interests include performance evaluation in mobile communications systems, WSN and Internet-of-Things, cognitive communications and networking, 5G mmWave cellular systems, and underwater communications and networks. He is currently Editor in Chief of the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING, and was Editor-In-Chief of IEEE WIRELESS COMMUNICATIONS from 2003 to 2005 and of the IEEE TRANSACTIONS ON COMMUNICATIONS from 2008 to 2011. He served as a Member-at-Large of the Board of Governors of the IEEE Communications Society from 2009 to 2011, and as its Director of Education in 2014-15.