TED: A Topology Discovery Algorithm for Underwater Acoustic Networks

Roee Diamant, Roberto Francescon and Michele Zorzi
Department of Information Engineering, University of Padova, Italy
Email: roeed@univ.haifa.ac.il, frances1@dei.unipd.it, zorzi@dei.unipd.it

Abstract

We describe TED, the first designated topology efficient discovery algorithm for underwater acoustic networks. Topology information is essential for network operations. By knowing the network topology, information sources can determine destination nodes and routing possibilities for their packets and schedule transmissions accordingly. It is therefore of interest to use a schedule for topology discovery to be used at the early stages of network operation. Such schedule must efficiently perform topology discovery and also guarantee a convergence time at the end of which topology discovery is complete and the network switches to its steady-state scheduling protocol. Considering this need, we offer a topology efficient discovery (TED) algorithm aimed to discover acoustic links and to assess their reliability. Designed to reduce the time overhead posed by the topology discovery phase, TED allows nodes to share time slots while controlling the number of possible collisions such that the delay of the topology discovery process is minimized. TED also discovers scheduling conflicts by discovering node pairs whose transmissions block one another, often referred to as near-far node pairs (NFNPs). Information about NFNPs can assist for power control or to increase channel utilization if interference cancellation techniques are employed. TED is applicable for underwater networks consisting of modems whose transmission range is on the order of a few km and above, and when the maximal distance between the nodes is known to be large or assumed equal to the modem's transmission range. Numerical results show that under these conditions, TED accurately detects the network topology in a much shorter time compared to benchmark methods. The results also show that, using TED, more packets are received, and thus the accuracy in determining the reliability of communication links increases. We also report results from a sea experiment, where TED was tested in different sea environments.

Index Terms

Topology Discovery, Underwater Networks, Near-Far, Spatial Reuse, Packet Collisions, Acoustic Communication

Parts of this work have been presented at the Underwater Communications conference (UComms2016).

I. INTRODUCTION

Underwater acoustic networks (UWANs) include fixed or mobile nodes whose task is to collect data and report it, usually to a common sink but also to one another. Nowadays, UWANs are envisioned for many different applications, including marine data collection to monitor and forecast weather, environmental monitoring (e.g., water pollution or marine mammals), military underwater surveillance and security of offshore infrastructures, underwater navigation assistance and industrial applications (marine gas drilling). The operation of the network requires topology information to determine destination nodes, to allow routing and to schedule transmissions.

Although the number of nodes in typical UWAN deployments is only on the order of 5-20, recent works have shown that utilizing topology information may significantly improve the throughput of UWANs. By detecting nodes whose transmissions do not collide, nodes can share time slots to optimize channel utilization [1], [2], or a hierarchal tree structure can be formed to allow simultaneous transmissions [3], [4]. In fact, as shown in [5], topology information can be used to increase the throughput of UWANs even beyond that of terrestrial networks.

Network topology includes not only the list of all communication links in the network, but also the reliability of such links and the possible conflicts between them [1]. Reliability of communication links reflects the outage capacity of the link, and affects the availability of the network. In UWANs, the reliability of the link is a function of the multipath interference, the ambient noise, and the spatial-dependent power attenuation [6], and is directly connected to the network throughput [5]. Since in UWANs nodes tend to continuously move due to the water currents, assessing the link reliability is challenging [7]. Considering this problem, a common practice is to use multiple broadcast transmissions to discover communication links and to estimate their reliability. This is mostly done by measuring the rate of successful packets [8], [9], or by estimating the signal-to-interference plus noise ratio (SINR) through measuring the nodes inter-distances and using an attenuation model [10]. This process involves an initial collision-free scheduling protocol, mostly using time-division-multiple-access (TDMA), where nodes broadcast their information. However, often these initial collision-free transmissions are extremely long, and topology discovery imposes a large time overhead [11]. Instead, motivated by previous approaches for topology-transparent scheduling, we offer to employ spatial reuse already at the initial stage of topology discovery. Our solution leads to more packet transmissions per time unit, so that the time overhead is reduced and the link reliability can be determined more accurately.

Our solution, referred to as the topology efficient discovery (*TED*) algorithm, also discovers scheduling conflicts by detecting node pairs whose packets block (or jam) each other, often referred to as near-far node pairs (NFNPs). The near-far phenomenon occurs when there is a large difference in the received power of the two nodes. This can occur when the jammer is transmitting with much higher power, or when transmissions from the jammed node are highly attenuated (e.g., it is located below a sound layer or with no direct acoustic line-of-sight). Due to the large attenuation per unit distance in the underwater acoustic channel, near-far is common in UWANs also when the ratio between the jammer-receiver range and the jammed-receiver range is not so large [12]. NFNPs are usually treated either as a scheduling challenge [9], [13], or as a target for power control [14], or for multipacket reception (MPR) if interference cancellation techniques are employed [15]. NFNPs are often detected by evaluating the signal-to-interference ratio (SIR) from measurements of the signal-to-noise ratio (SNR) [9]. However, for NFNP detection, there is a clear benefit for an actual decoding attempt to discover a node whose packets are jammed by another. To that end, TED employs deliberate packet collisions. Then, if only packets originated by a single node are properly decoded, an NFNP is discovered.

The contribution of this paper is twofold and consists of:

1) the first designated algorithm for topology discovery in UWANs to reduce time overhead and to accurately evaluate link reliability, and
2) a novel algorithm to efficiently detect NFNPs.

TED is most effective underwater networks with modems whose transmission range is on the order of a few km and above, and when the maximal distance between the nodes is known to be large or assumed equal to the modem's transmission range. We note that this latter case is common practice in the deployment phase of underwater networks. Numerical results show that, under the above two scenarios, compared to benchmark methods TED significantly reduces the time overhead of topology discovery while allowing a better estimate of the links' reliability. Our results were validated in a sea experiment conducted from southern Spain to the west coast of Africa, where we demonstrated the use of TED in multiple sea environments.

The remainder of this paper is organized as follows. The related state-of-the-art is described in Section II. The system model and objectives are introduced in Section III. In Section IV, we describe in detail the structure of TED. Simulation results are presented and discussed in Section V, and results from the sea experiment are shown in Section VI. Conclusions are drawn in Section VII.

II. State-of-the-Art

In this section, we review previous approaches to acquire topology information, and methods to evaluate the link reliability.

A. Topology Discovery

Topology discovery is processed distributedly by nodes forming their sets of one-hop neighbors and lists of conflicting links. The state-of-the-art includes two main classes of scheduling solutions, namely, pre-determined and random. In random topology discovery (also refer to as ”contention-based”), the schedule is not set. Instead, nodes either transmit at random using some version of the Aloha protocol [16], [17], or follow a multiple-access-collision-avoidance (MACA) protocol to exchange RTS/CTS packets and discover links by overhearing control packets [18]. A collision-avoidance protocol that can be used for topology discovery is suggested in [19], where control packets are exchanged only at the beginning of set time slots. In [20], a combination of TDMA and MACA is proposed, where at the beginning of each time slot a node transmits only after sensing the channel. Alternatively, in [21] the protocol starts with MACA to discover immediate links. Then, time slots are assigned by passing a token to discover the tree hierarchy of the network. However, in networks with a limited number of nodes (which is the case of UWANs [11]), for the purpose of topology discovery where not only the number of received packets but also the reception consistency are of interest, a pre-determined schedule may be more appropriate.

Pre-determined scheduling is largely based on the concept of TDMA where, in each time slot, the identity of the transmitting nodes is pre-determined. TDMA scheduling can be divided into collision-avoidance and collision-permitted. The former is mainly based on simple TDMA, where each node receives a designated time slot to transmit [18]. The time frame then repeats until enough transmissions are made to assess the link reliability [9]. The TDMA time slot includes a guard interval, whose length is comparable to the duration of the information packet. This greatly limits the network throughput and impacts the time overhead of the topology discovery phase. Considering this problem, several methods suggested to shorten this delay by employing spatial and/or time reuse. In [3], nodes can 'steal' a time slot by identifying those in which transmissions are not overheard. For the same purpose, carrier sensing is performed in [3]. Alternatively, collision-permitted schedules allocate the same time slots to several nodes, while limiting the maximum number of packet collisions. Such is the method proposed in [22], where Galois field polynomials are used to determine the number and identity of nodes transmitting in each time slot, as well as the number of time slots in each time frame. For the case when the maximum degree of the network (i.e., the number of one-hop neighbors a node may have) is known, [23] formalized a constrained optimization problem to allow each node a minimum number of transmissions while limiting collisions to a target level. Further collisions are removed by measuring the collision rate and adapting the schedule accordingly [24]. Utilizing the fact that many nodes may have a much lower degree than the maximum degree of the network, an improvement is suggested in [25], where unassigned time slots are identified and re-allocated. Yet, this requires knowledge of two-hop neighbors, which may not be available in the topology-discovery phase, and a method to optimize the network performance already in the topology discovery phase is still required.

B. Link Reliability

Estimating the reliability of a communication link is an important objective in any communication system. A link between nodes should be established only if the communication between the nodes is considered reliable. In terrestrial networks, link reliability is determined numerically and is supported by a large database of measurements (e.g., see [26]). Unfortunately, such a large amount of data is not available for UWANs. Instead, the main performance metric is the packet error probability [27]. However, due to the time varying nature of the underwater acoustic channel and of the ambient noise [6], the packet error probability is affected by only an instantaneous notion of link reliability, and cannot reflect on the target quality of service.

To estimate the link reliability, several approaches use statistics gathered during the topology-discovery phase. In [28], a link is considered reliable if 85% of the packets scheduled in this initial phase are correctly received. Alternatively, in [29] the definition of a reliable link is based on the received signal power level as long as the packet is deemed correctly decoded using a successful cyclic redundancy check (CRC). Other works suggested measuring the link coherence time [30], or the channel delay spread [31] as both metrics affect the link reliability. However, currently, there is no unique definition of link reliability in UWANs, and consequently, no work offers a designated method to measure it.

III. Preliminaries

In this section, we introduce the system model and the objectives considered in this work. We also present our definition of link reliability.

A. System Model

Our system includes a set of N nodes, $\mathcal{N}$. The nodes do not hold topology information of any kind, except for the network size, N, and the capabilities of the acoustic modem in terms of the transmission range, T_c, as declared in the specifications[1]. Since underwater networks usually include a small number of homogeneous nodes and are deployed by a single operator, we argue that the above two assumptions are reasonable even when nodes are permitted to join or leave the network. The nodes form a UWAN and send information to one another via a pre-defined topology-discovery topology-transparent scheduling protocol. We assume the nodes are roughly synchronized such that it is possible to bound the offset between the nodes' internal clock by some value that is shorter than the maximum propagation delay, otherwise a preliminary time synchronization phase is required (e.g., [32], [33]). Our solution, TED, is based on time windows allocated for transmission. We use the following definitions: a *time slot* is defined as the time window during which a certain node is allowed to transmit, and may consist of the transmission of several (I) packets. A *time frame* is defined as the time window in which all nodes are given at least a single time slot for transmission. The *convergence time* of TED is defined as the time window until all links are discovered, and may consist of several time frames.

Our objective is to allow node $n \in \mathcal{N}$ to obtain topology information in the form of 1) the list of its entire one-hop links and their reliability, and 2) the set of all NFNPs with respect to itself. To that end, we let each node, i, broadcast packets so that each (unknown) one-hop neighbor node, j, who receives these packets would discover the link to it. Since transmissions are broadcast

[1]Please note that the transmission range, which is a distance, is equivalently expressed here for convenience in terms of the corresponding propagation delay.

and since at the phase of topology discovery the propagation delays are unknown, solutions that avoid collisions by means of time reuse (e.g., [34], [35]) are not possible. Instead, our solution is formed by an $(N-1) \times (N-1)$ topology matrix $\boldsymbol{M}(n)$ [2] whose entry $M_{k,j}(n) = p_{k,j}$ is the probability of successful decoding of packets from node k when node j is also transmitting. The diagonal elements of $\boldsymbol{M}(n)$ correspond to the probability of receiving interference-free packets, whereas the other entries represent the NFNP opportunities. Let $P_{\mathrm{error,req}}$ be a target packet error rate set by the user to define the maximum packet error rate for a link to be considered as reliable. Then, if $M_{k,j}(n) > 1 - P_{\mathrm{error,req}}$ and $M_{j,k}(n) < 1 - P_{\mathrm{error,req}}$ nodes (k, j) are an NFNP. In this case, j is the jammer node and k is the jammed node [3].

B. *Quality Measures*

For topology discovery, we are interested in finding stable links that remain reliable for a sufficient time. That is, it is not sufficient to set a threshold on the number of received packets on a link, but there is also a need to test the stability of the link over time. Considering the time-varying characteristics of the channel, we define link stability by the regularity of packet reception, which we refer to as a *link test*. We define a link test as a pair of packets successfully received less than $2T_{\mathrm{c}}$ s but more than T_{c} s apart. To count link tests, we divide time into windows of duration $2T_{\mathrm{c}}$ s [4]. Then, a link test is counted if in a certain time window there exists a pair of packets that satisfies the above condition. To determine a stable link, we require at least L link tests.

We measure the performance of TED in terms of four quality measures, namely, 1) the duration required until all topology information is obtained, 2) the per-node average number of link tests performed for link detection, 3) the per-node number of total packets received, and 4) the number of detected NFNPs. The first metric directly impacts the time overhead of the topology-discovery

[2] Throughout this paper, we denote a vector by a bold lower case letter, a matrix by a bold upper case letter, and matrix/vector entries with subindices.

[3] Throughout this paper, we detect and quantify interference between two nodes only. We note that interferences from two or more nodes create ambiguity and the method would be prone to errors, which, if not identified, may have a large impact on the transmission scheduling protocol. Hence, while detecting the case of three or more mutual interferences would allow further utilization of the NFNP phenomena, this case is not considered in TED.

[4] Note that this time windowing is performed by each node distributedly and does not require cooperation or time-synchronization.

phase. The second and third measures are related to the identification of communication links. That is, the level of accuracy in estimating the link reliability increases with the number of link tests and received packets. Compared with the true number of NFNPs, the fourth measure gives the accuracy of detecting NFNPs. Note that we do not consider the communication overhead of the topology discovery schedule. This is because topology discovery serves only to initialize the UWAN and thus its objectives are accuracy and time overhead rather than efficiency.

The process of topology discovery ends when all nodes complete at least L link tests (the sharing of the discovered topology is not within the scope of this paper). Hence, to formalize the first measure for the convergence time of the topology discovery, let $t_{n,m,i}$ be the time instant the ith link test of node n is successfully performed by its one-hop neighbor node m. The convergence time is:

$$\rho_{\mathrm{time}} = \max_{n,m} t_{n,m,L} \ , \tag{1}$$

Note that since topology discovery is performed distributedly, the minimization of the time elapsed until the protocol converges is performed on a per-node basis. Let $y_{n,m}^{\mathrm{link}}$ be the number of link tests made for the communication link between node n and node m. Then, the second metric that counts the average number of link tests is

$$\rho_{\mathrm{link}} = \frac{1}{N(N-1)} \sum_{n=1}^{N} \sum_{\substack{m=1 \\ m \neq n}}^{N} y_{n,m}^{\mathrm{link}} \ . \tag{2}$$

For the third measure, we count the number of received packets. Let $y_{n,m}^{\mathrm{pck}}$ be the number of packets transmitted by node n and successfully received by node m during ρ_{time} from (1). Measure three is formalized as

$$\rho_{\mathrm{pck}} = \frac{1}{N(N-1)} \sum_{n=1}^{N} \sum_{\substack{m=1 \\ m \neq n}}^{N} y_{n,m}^{\mathrm{pck}} \ . \tag{3}$$

Finally, to formalize the fourth quality measure let y_{n}^{NFNP} be the number of NFNPs accurately detected by node n during ρ_{time} from (1). Then, we measure

$$\rho_{\mathrm{NFNP}} = \frac{1}{N} \sum_{n=1}^{N} y_{n}^{\mathrm{NFNP}} \ . \tag{4}$$

C. Definition of Link Reliability

Let $P_{\text{error,req}}$ be the target packet error probability of the communication system, and $\mathrm{P}_{\text{error,act}}$ be the measured one. We define link reliability as

$$\mathrm{P}_{\text{reliab}} = \mathrm{P}_{\text{r}}(\mathrm{P}_{\text{error,act}} \leq \mathrm{P}_{\text{error,req}}) \ . \tag{5}$$

To calculate $\mathrm{P}_{\text{reliab}}$ from (5), the cumulative distribution function (CDF) of $\mathrm{P}_{\text{error,act}}$ is needed. Since this information is not available in the initial topology discovery phase, we simplify (5) by taking into account only the measured SINR [36]. That is,

$$\mathrm{P}_{\text{reliab}} = \mathrm{P}_{\text{r}}(\mathrm{SINR} \geq \mathrm{H}) \ , \tag{6}$$

where H is the SINR value that corresponds to $P_{\text{error,req}}$, and (6) is evaluated statistically from the received packets during topology discovery.

IV. Description of the TED Algorithm

In this section, we describe the details of our TED algorithm for topology discovery and NFNP detection. We start with the key idea behind TED. Then, we present its structure and discuss implementation details.

A. Key Idea

TED is a scheduling protocol designed to operate at the early stage of the network operation, and includes a fixed schedule that is uploaded to each of the nodes prior to the network deployment. The focus of TED is the initial discovery of communication links and NFNPs, and not to track topology changes. TED is performed distributedly and does not need nodes' cooperation but assumes a rough time-synchronization to allow nodes to start topology discovery with delay not exceeding T_{c}. TED consists of two algorithms for topology discovery: 1) a schedule for discovering communication links, and 2) a schedule for the joint discovery of communication links and NFNPs. The main idea in TED is based on the observation that the key objective is to reduce the time overhead introduced by the pre-schedule for topology discovery, and that many link tests are required in order to accurately evaluate the link reliability. TED therefore aims to transmit as many packets as possible while limiting the schedule's time frame.

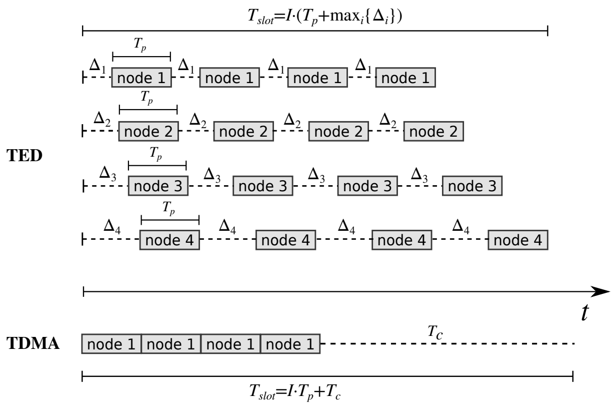
In both solutions, the reliability of each link is evaluated using (6) by measuring the SNR or SINR of each link.

TED pre-determines transmissions in a TDMA fashion. Towards the objective of link discovery only, the number of pre-determined nodes transmitting each I short packets of duration T_p is chosen such that the probability of collisions is minimized. To that end, TED delays the packets of each transmission node k ($k \in \mathcal{N}$) by a pre-determined optimized value, Δ_k. This delay value is minimized while limiting possible packet collisions and results in time slots of possibly non-equal durations $IT_p + I\max(\Delta_k),\ k \in \mathcal{N}$. In turn, the time frame of TED consists of a repetition of these time slots for different values Δ_k, and for different transmitting nodes' ID. The structure of a single time slot in TED for the link discovery phase is illustrated in Figure 1a. Note that a random choice of Δ_k will make TED a version of Aloha.

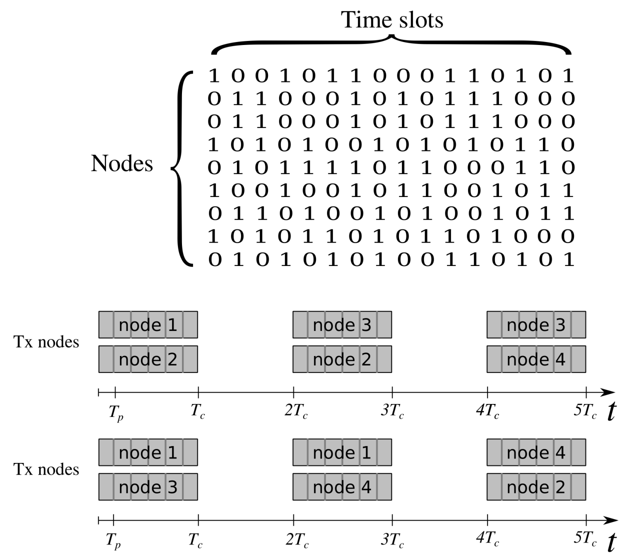
For the joint discovery of links and NFNPs, in each time frame, TED identifies a set of nodes and allows them to transmit a series of short packets. The transmissions are scheduled such that packet collisions are deemed to occur between each of the possible node pairs. Moreover, by sending a large number of consecutive short packets, TED is able to discover both communication links and NFNPs. To reduce the time overhead of the topology discovery process, the identity of the transmitting nodes in each time frame is determined such that the number of transmissions is maximized while limiting the maximum number of packet collisions. The structure of the time frame for the joint link discovery and detection of NFNPs of TED is illustrated in Figure 1b. TED is able to discover near-far interference between any node pair in the network. In the example given in Figure 1b, mutual interferences (if they exist) are discovered between node 1 and any of nodes 2, 3 and 4. However, TED is limited to the detection of NFNPs, and the jamming of one node due to interference from two or more nodes is not considered.

B. Link Discovery Only

In the link discovery phase, TED allows X nodes to share the same time slot and to transmit, each, I short packets of duration T_p, which include the ID of the transmitting node as well as a CRC. For simplicity, in the following we enumerate the X transmitting nodes by $x = 1, \ldots, X$. As illustrated in Figure 1a, the individual packets transmitted are delayed by a pre-assigned vector $\boldsymbol{\Delta} = [\Delta_1, \ldots, \Delta_X]$, where for a node $x \in \mathcal{N}$, Δ_x is minimized such that at least one collision-free transmission per communication link is guaranteed with high probability. Note that keeping



(a) Link discovery only (one time slot x)



(b) Joint link discovery and detection of NFNPs

Fig. 1: Illustration of time frames in TED.

$\mathbf{\Delta}$ fixed sets an organized structure through which a probability to detect communication links can be guaranteed. For a specific communication link, once L link tests (see (2)) are successful, the receiver includes the link in its topology information and sets the link's reliability according to (6).

TED alternates the identity of the X nodes sharing the same time slot. In each time frame, N/X transmission segments are scheduled. In the ith segment, nodes $(i-1)\cdot X+1,\ldots,i\cdot X$ are scheduled to transmit. During each time frame, TED allows discovery of all communication links. This is achieved by allowing a transmitter x to also attempt to decode packets of other nodes during its Δ_x packet delay period. In the following, we describe the process of determining the parameters of TED.

The advantage of TED over regular TDMA, even when we let each node transmit I packets in each time slot, is due to its many more transmissions in a given time. Instead of I transmissions per time slot, in TED $X \cdot I$ transmissions are issued per slot. Also, instead of a time slot of duration $(T_{\rm c}+I\cdot T_p)$ s and the need for NL time slots until convergence in TDMA, in TED the duration of the time slot is $I(T_p + \max_x(\Delta_x))$ and it takes LN/X time slots until convergence. Thus, compared to regular TDMA where in total $NL{\cdot}I$ packets are transmitted in the convergence time of duration $L \cdot N(T_\mathrm{c} + I \cdot T_p)$ s, in TED for the same time duration the total number of transmitted packets is

$$N_{\text{packet}} = X \cdot I \cdot \frac{LN \cdot (T_\mathrm{c} + I \cdot T_p)}{I(T_p + \max_x(\Delta_x))} \ . \tag{7}$$

For example, considering $N = 8$, $X = 4$, $I = 5$, $L = 4$, $T_\mathrm{c} = 2$ s, $T_p = 0.05$ s, and $\max(\Delta_x) = 3T_p$, the number of packets transmitted in TED is nine times more than in TDMA. In fact, TED employs more than twice as many packets as TDMA, starting from $T_\mathrm{c} = 0.25$ s, which implies that TED is suitable also for small networks. Although some of these packets will collide (as opposed to TDMA where all receptions are interference free), still many more interference-free packets are received in TED.

1) Determining the Parameters of Link Discovery:

The process of determining $\boldsymbol{\Delta}$ is performed a priori. To select Δ_x, we observe that two metrics trade off. On one hand, to reduce the time overhead of TED, we are interested in decreasing the time frame. On the other hand, a long time frame would help to mitigate packet collisions. To control this trade off, we determine the parameters of the link discovery phase by solving a min-max problem to optimize a combination of the channel utilization and the packet success rate. To achieve both objectives, we set Δ_x in the inner part of the optimization problem, while its upper limit is determined in the outer part. Therefore, besides parameters, I, X, and Δ_x, we introduce another parameter, τ, and set the upper limit as $\Delta_x \le \tau \cdot T_p$. This upper limit cannot exceed T_c, otherwise TED would have no advantage over TDMA, and cannot be below T_p, otherwise TED would become inefficient. Parameter τ is therefore set between 1 and $\lfloor T_\mathrm{c}/T_p \rfloor$.

For $N = 8$ nodes located in uniformly random locations within a square area of 2000×2000 m^2 for $\tau = 3$, Figure 2 shows a histogram of the number of packet collisions for a set of 200 different permutations of vector $\boldsymbol{\Delta}$. We observe that the number of packet collisions dramatically changes, and therefore conclude that the values in vector $\boldsymbol{\Delta}$ heavily affect the number of collisions. Motivated by this result, in the following we find the values for $\boldsymbol{\Delta}$ and τ, such that the probability of collision is minimized.

Let $T_\mathrm{c}^{(k,r)}$ be the (unknown) propagation delay of link (k, r). With respect to receiver r, the ith packet $(i = 1, \dots, I)$ from node k $(k = 1, \dots, X)$ may collide with the i'th $(i' = 1, \dots, I)$

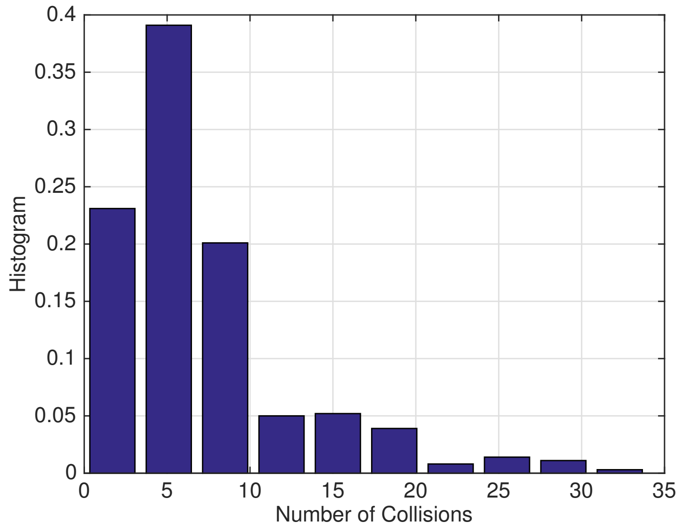


Fig. 2: Histogram of the number of collisions for randomized values of the delay vector $\boldsymbol{\Delta}$. Since the number of collisions changes dramatically, we conclude that there is a strong dependence on the number of collisions on $\boldsymbol{\Delta}$.

packet of node x ($x = 1, \ldots, X$, $x \neq k$) if

$$\begin{aligned} T_{\mathrm{c}}^{(k,r)} + (i-1)\cdot(T_p + \Delta_k) + \Delta_k &< T_{\mathrm{c}}^{(x,r)} + (i'-1)\cdot(T_p + \Delta_x) + \Delta_x \ \text{ and} \\ T_{\mathrm{c}}^{(k,r)} + (i-1)\cdot(T_p + \Delta_k) + T_p + \Delta_k &> T_{\mathrm{c}}^{(x,r)} + (i'-1)\cdot(T_p + \Delta_x) + \Delta_x \ , \end{aligned} \tag{8}$$

or if

$$\begin{aligned} T_{\mathrm{c}}^{(k,r)} + (i-1)\cdot(T_p + \Delta_k) + \Delta_k &< T_{\mathrm{c}}^{(x,r)} + (i'-1)\cdot(T_p + \Delta_x) + T_p + \Delta_x \ \text{ and} \\ T_{\mathrm{c}}^{(k,r)} + (i-1)\cdot(T_p + \Delta_k) + T_p + \Delta_k &> T_{\mathrm{c}}^{(x,r)} + (i'-1)\cdot(T_p + \Delta_x) + T_p + \Delta_x \ . \end{aligned} \tag{9}$$

Let $\mathcal{E}_1(k, x, i, i')$ represent event (8) and $\mathcal{E}_2(k, x, i, i')$ represent event (9).

For a random i.i.d. deployment of nodes within a square area, $T_{\mathrm{c}}^{(k,r)}$ is a random variable. However, without prior knowledge of the nodes' locations, we are not aware of the distribution for $T_{\mathrm{c}}^{(k,r)}$. Instead, we assume $T_{\mathrm{c}}^{(k,r)}$ to be uniformly distributed between 0 and T_{c}. Further below (see Figure 5), we show that even when the actual distribution of $T_{\mathrm{c}}^{(k,r)}$ is Rayleigh, the assumption of uniform distribution leads to a choice of Δ_k that is better than any random choice, and similar results were obtained for some other forms of distributions. For $\rho(i, k) = (i-1)\,(T_p + \Delta_k) + \Delta_k$, we obtain

$$\begin{aligned}\mathrm{Prob}\left(\mathcal{E}_1(k,x,i,i')\right) &= \frac{\rho(i',x)-\rho(i,k)+T_\mathrm{c}}{2T_\mathrm{c}}\cdot\frac{T_\mathrm{c}-\rho(i',x)+\rho(i,k)+T_p}{2T_\mathrm{c}}\ ,\\ \mathrm{Prob}\left(\mathcal{E}_2(k,x,i,i')\right) &= \frac{\rho(i',x)-\rho(i,k)+T_\mathrm{c}+T_p}{2T_\mathrm{c}}\cdot\frac{T_\mathrm{c}-\rho(i',x)+\rho(i,k)}{2T_\mathrm{c}}\ .\end{aligned} \tag{10}$$

Note that to allow a feasible solution for (10), $\rho(i',x)-\rho(i,k)$ must be bigger than $-T_\mathrm{c}$ and smaller than T_c. Given the maximum value allowed for Δ_x (in our case, $\tau\cdot T_p$), this constraint provides the maximum possible value for the number of transmitted packets in the time slot, I.

Besides $\hat{I}$, $\hat{\boldsymbol{\Delta}}$ and $\hat{\tau}$, selecting X poses a tradeoff between packet success rate and channel utilization. Since communication is half duplex, the former is measured by $(N-X)/(N-1)$, and the latter by X/N. However, since X affects the success rate of link detection, together with the other system parameters, it should be optimized using a single quality metric. To determine $\boldsymbol{\Delta}$, I, τ, and X, we minimize (10). Define

$$p_x(\boldsymbol{\Delta},I)=\left(1-\prod_{\substack{k=1\\k\neq x}}^{X}\prod_{i=1}^{I}\prod_{i'=1}^{I}\left[\mathrm{Prob}\left(\mathcal{E}_1(k,x,i,i')\right)+\mathrm{Prob}\left(\mathcal{E}_2(k,x,i,i')\right)\right]\right)\ .$$

Then,

$$\left[\hat{I},\ \hat{\boldsymbol{\Delta}},\ \hat{\tau},\ \hat{X}\right]=\arg\min_{I,\tau}\left(\arg\max_{\boldsymbol{\Delta},X}\frac{X}{N}\frac{N-X}{N-1}\sum_{x=1}^{X}p_x(\boldsymbol{\Delta},I)\right)\ , \tag{11a}$$

$$\text{s.t. } p_x(\boldsymbol{\Delta},I)\geq p_\mathrm{req},\ \ x=1,\ldots,X\ , \tag{11b}$$

$$0\leq\Delta_x\leq\tau T_p,\ \ x=1,\ldots,X\ , \tag{11c}$$

$$1\leq\tau\leq\frac{T_\mathrm{c}}{T_p}\ , \tag{11d}$$

where p_req is a user defined target probability to detect a communication link. To solve (11), we use the alternating maximization method to maximize (11) for an alternating subset of parameters while fixing the others [37]. We start the process by optimizing X while fixing $\tau=1$, $\Delta_x=T_p\cdot(x-1)/(X-1)$, and $I=5$. Other choices for initialization will lead to similar performance, yet this choice has led us to the shortest convergence time of (11). Since problem (11) is convex, this method is guaranteed to converge to the global optimum [37]. We note that (11) can be generalized by allowing $\boldsymbol{\Delta}$ to become a matrix whose elements $\Delta_{x,i}$ represent the delay for node x and for packet i. While this generalization offers an additional degree of freedom, it

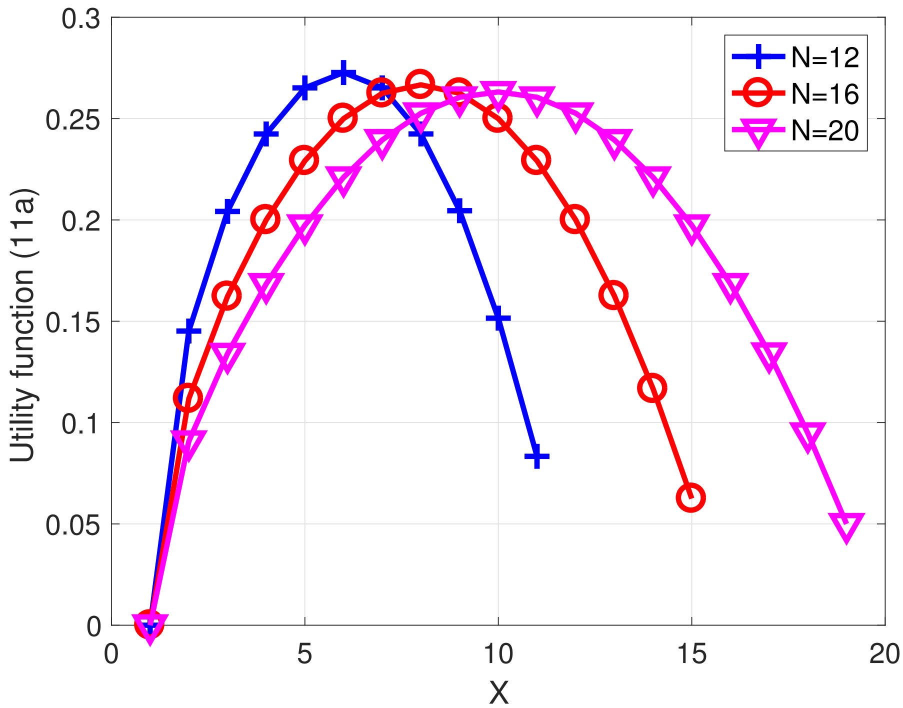


Fig. 3: Utility function (11a) as a function of the number of transmitting nodes X for three values of N. The figure shows that the optimal X is obtained for $N/2$.

significantly complicates the solution as many more parameters would need to be optimized. We therefore leave this extension for future work.

Solving (11), we determine the number of nodes allowed to transmit in each time slot to be $\hat{X} = \left\lfloor \frac{N}{2} \right\rfloor$. This is observed in Figure 3, where we show the value of utility (11a) as a function of X for three values of the number of nodes, N. Furthermore, while for the solution vector $\hat{\boldsymbol{\Delta}}$ from (11) the values of $\mathrm{Prob}\left(\mathcal{E}_1(k,x,i,i')\right)$ and $\mathrm{Prob}\left(\mathcal{E}_2(k,x,i,i')\right)$ are expected to be small, there may still be packets colliding. For this reason, in each time frame, TED reassigns nodes with different delay values from the solution vector $\hat{\boldsymbol{\Delta}}$. As a result, the probability of packet collision will change across different time frames, and the probability that at least one packet is received by the one hop neighbors of each node increases.

C. Joint Link Discovery and Detection of NFNPs

TED offers a mechanism for joint link discovery and detection of scheduling conflicts. While for the former goal interference free transmissions are required, for detection of conflicts the schedule should guarantee collisions. The solution is based on the observation that detection of NFNPs is possible only when exactly two packets collide, while a collision that involves three or more packets creates ambiguity. Furthermore, since the propagation delay is assumed uniform between 0 and T_{c}, to guarantee packet collision, the transmission time of each of the two transmitting nodes should be no less than T_{c}. Considering these observations, for both detection of communication links and detection of NFNPs, TED schedules two nodes to share the time slot and to transmit for T_{c} seconds. Since false detection of NFNPs may be catastrophic for the network, TED is conservative and limits the number of nodes transmitting in the same time slot to 4. This way, in each time frame, TED can detect up to six possible NFNPs. The identity of these four nodes is determined such that the number of transmissions is minimized while limiting the maximum number of packet collisions as discussed in Section IV-C2 further below.

1) Structure of time slot:

As illustrated in Figure 1b for the example of $N = 8$, each node $i_1, \ldots, i_4$ transmits a series of short packets of duration T_p. The time slot is divided into two segments. In the first segment we discover possible communication links and NFNPs between pairs (i_1, i_2), (i_2, i_3), and (i_3, i_4). In the second segment, we discover links and possible NFNPs between pairs (i_1, i_3), (i_1, i_4), and (i_2, i_4). At the beginning of the first segment, node i_1 transmits $L = T_{\mathrm{c}}/T_p$ packets and at the same time node i_2 also transmits $L = T_{\mathrm{c}}/T_p$ packets. If i_1 and i_2 share a common receiver and since any propagation delay in the network is bounded by T_{c}, this schedule ensures that some packets will surely collide but a few packets from each node will still be decoded to allow link discovery. Next, $2T_{\mathrm{c}}$ seconds after the beginning of the time slot, nodes i_2 and i_3 transmit $L = T_{\mathrm{c}}/T_p$ packets. This delay ensures that nodes i_1 and i_3 will not collide. Last, T_{c} seconds after the last transmission of node i_2, nodes i_3 and i_4 transmit. As we visualize in Figure 1b, in the second segment we reverse the order of transmissions. At the beginning of the segment, we let nodes i_3 and i_1 transmit. Then, we let nodes i_1 and i_4 share the channel. Last, we let nodes i_4 and i_2 transmit together $4T_{\mathrm{c}}$ seconds after the beginning of the segment. The two segments end with a guard time of T_{c} seconds, such that the total time duration of the time slot is $12T_{\mathrm{c}}$ seconds.

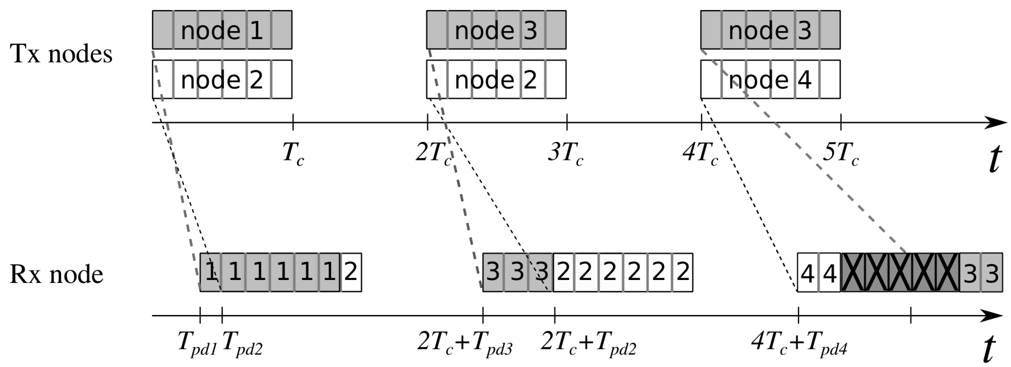


Fig. 4: Example for the detection of NFNPs. Nodes $(2, 1)$ and $(3, 2)$ are NFNPs, while nodes $(3, 4)$ are not.

To determine if a node pair is an NFNP, we consider the following two situations:

1) NFNPs form sequences of reception. That is, if node i jams node j (and thus its packets will commonly arrive before those of node j), there will be a sequence of packet decodings from node i followed by a sequence of packet decodings from node j. This is illustrated in Figure 4.
2) If the propagation delay of the NFNPs is about the same but one node transmits with much higher power, many more packets originated from the jamming node will be decoded than from the jammed node.

Considering these observations, we set two threshold levels: Th_1 which is the expected number of decoded packets, and Th_2 which determines the expected length of a sequence of non-decoded packets due to channel noise. Clearly, $Th_1 = L \cdot (1 - \mathrm{P}_{\text{error,req}})$ and Th_1/L is the expected ratio of received packets. Furthermore, given the expected packet error rate due to channel noise, $\mathrm{P}_{\text{error,req}}$ from (5), we extract Th_2 from

$$\mathrm{P}_{\text{error,req}}{}^{Th_2} = \mathrm{P}_{\text{error,seq}} \ , \tag{12}$$

where $\mathrm{P}_{\text{error,seq}}$ is the expected probability of a sequence of Th_2 erroneous packets, which can be user defined (we use $\mathrm{P}_{\text{error,seq}} = 10^{-5}$) or determined from the expected duration of long noise transients (e.g., waves).

The pseudo-code for the detection algorithm is shown in Algorithm 1. The algorithm works in chunks of $2T_{\text{c}}$ seconds, where in each chunk only two nodes, i_1 and i_2, can collide. Denote $T_{x,1}$

as the time instant the first packet from node x is received, and $T_{x,2}$ as the time instant the last packet from node x is received. Also let N_x be the number of packets successfully received from node x. Assuming that the actual packet error rate is smaller than the required one, $\mathrm{P}_{\mathrm{error,req}}$, we test whether nodes i_1 and i_2 are NFNPs by testing if condition (8) or (9) applies (line 3). Then, to identify the jammer node, we consider the case where the propagation delay of the link to node i_1 is about the same as that of the link to node i_2. Here we consider i_1 as the jammer if its number of received packets is much larger than that of node i_2 (lines 4-5). For the case of distinct propagation delays, we check the ratio between the number of decoded packets from i_1 and i_2 (lines 6-7). Last, we consider the case where packets from i_2 are never received due to interference from node i_1 (lines 10-11). The algorithm then repeats itself to test whether i_2 is jamming i_1 (line 13). In the example illustrated in Figure 4, nodes $(2, 1)$ and nodes $(3, 2)$ are NFNPs and a proper detection is made by testing the number of received packets (lines 4 and 6 in Algorithm 1). However, nodes $(3, 4)$ are not NFNPs and indeed the conditions set in Algorithm 1 do not apply.

Algorithm 1 The NF-TDMA algorithm

1: Inputs: $T_{i_1,1}, T_{i_1,2}, T_{i_2,1}, T_{i_2,2}$ and N_{i_1}, N_{i_2} .
2: {Check consecutive sequences of received packets}
3: **if** ($|T_{i_1,2} - T_{i_2,1}|/T_p < Th_2$) **then**
4:     **if** ($N_{i_1} > Th_1 \cap N_{i_2} < Th_1$) **then**
5:         set i_1 as jammer of i_2 .
6:     **else if** ($N_{i_1} > Th_1 \cap N_{i_2} > Th_1 \cap \frac{N_{i_2}}{N_{i_1}} < Th_1/L$) **then**
7:         set i_1 as jammer of i_2 .
8:     **end if**
9: **end if**
10: **if** ($T_{i_2,1} = \infty$) **then**
11:     set i_1 as jammer of i_2 .
12: **end if**
13: Return to line 3, and switch i_1 with i_2 .

2) Identity of the Transmitting Nodes:

To determine the identity of the $y = 4$ node transmitting in the same time slot, we find the number of combinations q such that all communication links are explored at least once. One solution to this combinatorial problem is based on the concept of Latin squares. Latin squares are $X \times X$ square matrices of unique rows such that each value in the matrix appears exactly once in each row and in each column. We desire to schedule enough transmissions such that collisions between each pair of nodes are tested the minimum possible number of times but at least once. Using polynomial theory [22], the minimum number of Latin squares required to guarantee such an event is $q = \lceil \frac{1}{2} + \sqrt{\frac{1}{4} + N} \rceil$. Then,

$$s = q \cdot 4 \tag{13}$$

time slots are set. Following the procedure described in [22], for a node n, TED finds the column j_x for which n modulo y appears in the x row of the $\lceil n/y \rceil$ Latin square. TED then assigns node n to transmit in time slot $(x-1) \cdot y + j_x,\ x = 1, \ldots, q$. Recall that, for TED, we only need to find one solution for the above transmission assignment problem. By choosing y to be a prime power, we guarantee that at least one such solution can be found. In Figure 1b, we demonstrate this process for $N = 8$ nodes.

V. Simulation Results

In this section, we describe the performance of TED in terms of ρ_{time} from (1), ρ_{link} from (2), ρ_{pck} from (3), and ρ_{NFNP} from (4). We recall that TED is operated in the first stage of the network for the purpose of topology discovery. As such, the operation of TED is bounded by the assumed convergence time, and from that moment on TED is replaced by the steady-state MAC protocol of the network. Recall that to consider link stability, we measure this convergence time as the time it takes until at least $L = 4$ link tests are successfully made for each link, where link tests correspond to packets successfully received less than $2T_{\text{c}}$ but more than T_{c} s apart.

Our setting includes N nodes which exchange short packets of 8 bits (3 bits for node ID + 5 bits for CRC). Considering the need for synchronization signals and a possible training sequence, extra overhead symbols are added such that the overall packet duration is $T_p = 50$ ms. A Monte-Carlo set of 1000 topologies is generated. For each topology, nodes are uniformly randomly placed in a square area of 2000×2000 m^2 such that the maximum propagation delay is $T_{\rm c} = 1.88$ seconds. Note that a node may not receive a packet not only if the packet collides, but also if it is simply not connected to the transmitter. To increase network sparsity, we include four horizontal obstacles and one vertical obstacle uniformly randomly placed within the square area, with lengths uniformly distributed in $[50, 100]$ m, and consider communication links between nodes with an acoustic line-of-sight.

Since we focus on the performance of the scheduling algorithm, we ignore some channel effects like the Doppler shift and the Doppler spread. However, the effect of the multipath channel is considered through the calculation of the SINR for each received packet. To that end, for each node pair with an acoustic line-of-sight, we run the Bellhop model [38], which provides the full phase multipath ray-tracing model for the channel impulse response. We then calculate the packet SNR as the ratio between the power of the direct path and the sum of the channel noise with the power of the other multipath arrivals. For Bellhop, we consider shallow water of depth 100 m, flat sand bottom, a fixed sound speed of 1500 m/s, a carrier frequency of 10 kHz, a source level SL (SL is a parameter) and a noise level of 40 dB re 1μPa/Hz. In case two or more packets are simultaneously received, the SNR level of each packet is used to calculate the SINR level of each packet. Using the SNR (or SINR) level and considering, for simplicity, BPSK communication with no channel coding, we calculate the packet error rate $\mathrm{P}_{\rm error,act}(n, m)$ of link (n, m), and accordingly generate packet errors randomly. To measure the performance in different channel conditions, we use different source levels, SL=$\{170, 160, 155\}$ dB re 1μPa at 1 m such that the expected packet error rate at the maximum transmission range is $\mathrm{P}_{\rm error,max} = \{0.05, 0.13, 0.47\}$.

Unfortunately, we did not find a proper benchmark system for the detection of NFNPs. Instead, we compare the NFNPs detection performance of TED with the ground truth. To compare the topology-discovery performance of TED, we consider as benchmark both pre-determined schedules (as in TED) and high throughput random access schedules. For the former, we choose the simple time frame (*TDMA*), where node n transmits only in time slot n and there exist a total of N time slots per time frame. Another considered benchmark is the topology-transparent protocol suggested in [22] (*TT-TDMA*) in which several nodes share the time slot and collisions are permitted. While the concept of TT-TDMA is also used in TED for the choice of the transmitting nodes, the two protocols are fundamentally different. The main difference between TED and TT-TDMA lies in the structure of the time slot, which for TT-TDMA is the same as in TDMA while in TED it does not include guard intervals. This allows TED to schedule more than two transmitting nodes per time slot (as in TT-TDMA) while still keeping a maximum of one collision per node pair.

For random access benchmark schedules, we choose the basic ALOHA (*Aloha*) protocol whose high throughput when topology is unknown makes it appealing for link discovery, and a slotted ALOHA (*Slotted Aloha*) protocol which is heavily used in underwater communications. In the former, a node transmits immediately upon receiving a packet and retransmits after a random backoff time in case of packet collisions [39]. In the latter, the nodes transmit only at the beginning of time slots [17]. Since in the topology discovery phase, both the receiver and the transmitter are unaware of their one-hop neighbor nodes, we make a modification in Aloha and allow retransmissions if a node overhears a "NACK" message not more than $T_{\rm c}$ s after it transmitted a packet. This modification is needed to avoid the case of unnecessary packet retransmissions which increase packet collisions and impact the performance of ALOHA. More specifically, without a NACK, it is not possible for a node to decide whether its packet has collided and retransmission is needed or simply there is no communication link. As an advantage to ALOHA, in our simulations we assume NACK packets always arrive. While, assuming 50% load, for $N = 8$ the optimal backoff of ALOHA is 1.25 packets/node, we choose the backoff time based on simulation testing. This is because 1) we do not assume a fully connected network, and 2) we are interested in minimizing the convergence time (set by the number of link tests) which is harder to analyze. Based on a simulation study, the backoff time was chosen to be a random number uniformly distributed between 0 and Max-backoff=5 s. For Slotted Aloha, we follow the analysis in [17] and include in the time slot a guard interval of size 69% of the maximum possible propagation time $T_{\rm c}$. In Slotted Aloha, we allow a node to transmit with probability $1/b$ where b is the number of retransmissions for the considered packet.

A. Results for Link Discovery Only

Recall that for link discovery only, TED employs a time frame with two slots. Each slot includes $N/2$ nodes, each transmitting I packets. We consider $N = 8$ and $p_{\rm req} = 0.99$ (see (11)). For $T_{\rm c} = 1.88$ seconds and using (11), the result of the optimization becomes $\hat{I} = 5$, $\hat{\tau} = 3$, and $\hat{\mathbf{\Delta}} = [0, 0.031, 0.091, 0.1460]$. In this solution packet collisions will surely occur since $\Delta_x < T_p$ for $x = 1, 2$. Yet, since our objective is to have at least one collision-free reception in each time slot, such collisions are allowed. That is because, for example, since node $x = 1$ whose

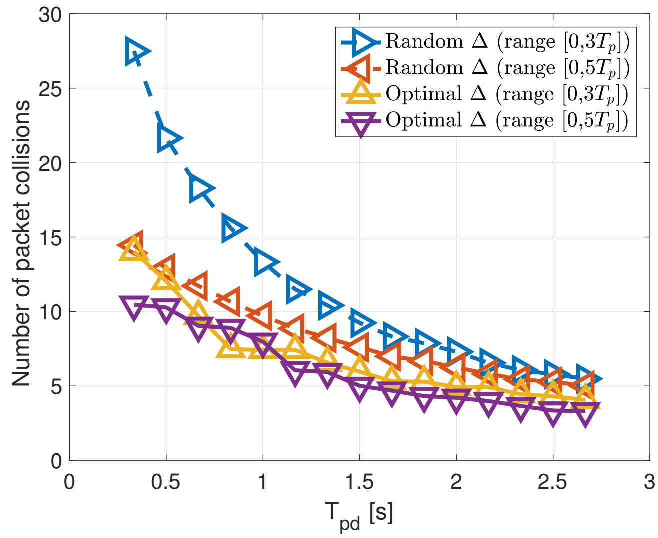
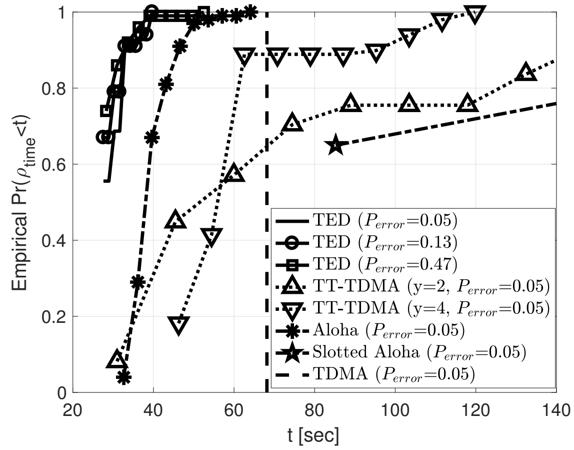


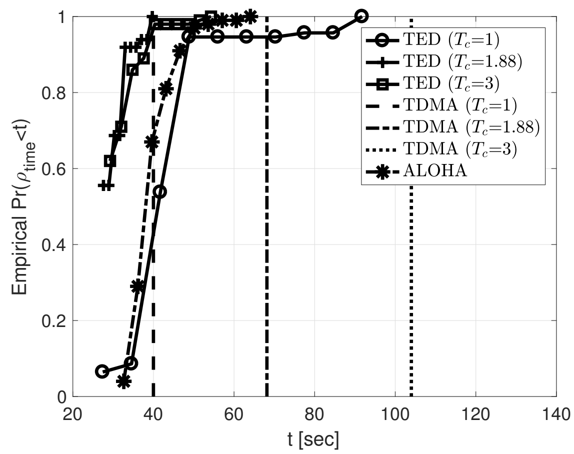
Fig. 5: Average number of packet collisions using the best of 500 random choices of $\boldsymbol{\Delta}$ and using $\hat{\boldsymbol{\Delta}}$ from (11) for two ranges of propagation delays. Propagation delays are chosen from a Rayleigh distribution. The figure shows a large benefit when using the calculated delay, especially for small networks

$\Delta_x = 0$ s would stop transmitting before node $x = 4$ whose $\Delta_x = 0.146$ s, packets of node $x = 4$ would be received without collisions from node $x = 1$ and vice versa.

We start with exploring the benefit of optimizing Δ_x. To that end, we compare the performance of TED in terms of the average number of packet collisions for $\boldsymbol{\Delta}$ chosen to be $\hat{\boldsymbol{\Delta}}$ compared to a simpler version of TED where the elements of $\boldsymbol{\Delta}$ are set randomly based on an i.i.d. uniformly distribution. For the latter, we show the results for the best out of a set of 500 i.i.d. randomized permutations. We note that this random version of TED is different than ALOHA since, unlike ALOHA, it pre-determines the number of transmitted packets as well as the number of transmitting nodes and does not require acknowledgments. The comparison is made for a set of 1000 random topologies. For a fair comparison, we use the same parameters for the optimized TED protocol and the random TED one. Specifically, since the optimized TED uses $\hat{\tau} = 3$, in the randomized TED protocol we let each element in $\boldsymbol{\Delta}$ be uniformly distributed in $[0, 3T_p]$ and in $[0, 5T_p]$. The results are shown in Figure 5 as a function of T_{c}. As expected, the results show that consistently using $\hat{\boldsymbol{\Delta}}$ gives a significantly lower number of packet collisions. We also



(a)

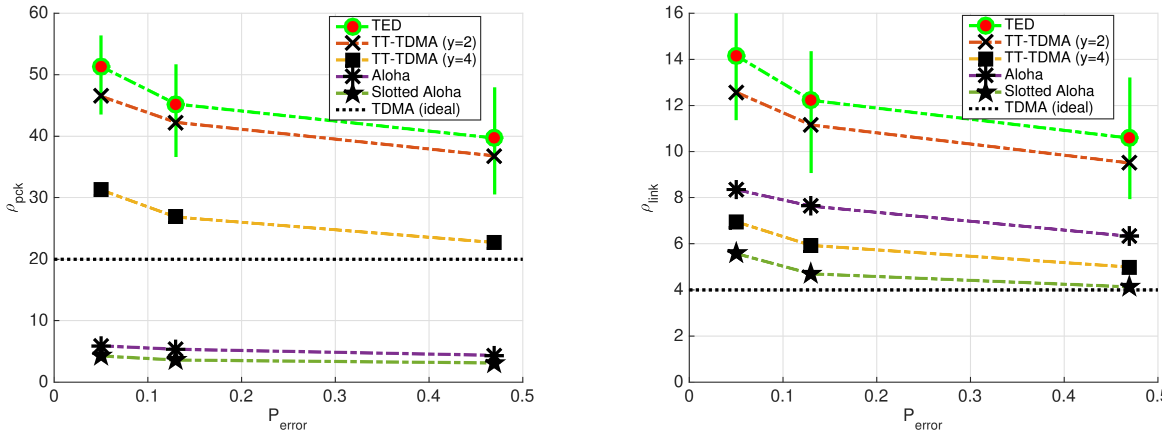


(b)

Fig. 6: Empirical CDF of ρ_{time} from (1) for: (a) different $\mathrm{P}_{\text{error,max}}$ values for $T_{\text{c}} = 1.88$, (b) different T_{c} values for $\mathrm{P}_{\text{error,max}} = 0.05$. Results shown for $N = 8$, $L = 4$, and $T_p = 50$ ms. Results show that, for $T_{\text{c}} > 1$ s, the convergence time of TED is shorter than that of TDMA, Aloha, Slotted Aloha, and both versions of TT-TDMA.

observe that this gain decreases with T_{c}. This is because for large propagation delay the number of collisions reduces, and the use of the optimized $\hat{\boldsymbol{\Delta}}$ has less effect. We note that a similar effect will occur if we increase $\hat{\tau}$, i.e., the range of possible values for Δ_x. However, this extension will be at the cost of larger delay in the topology discovery.

Recall that we measure the performance in terms of the convergence time of the process of link discovery, ρ_{time} from (1), defined as the maximum time it takes to detect at least $L = 4$ packets whose transmissions are spaced in time by at least T_{c}. Clearly, using TDMA and assuming, for fairness in comparison, the transmission of $\hat{I} = 5$ packets of duration T_p in each slot, the convergence time of TDMA is $4N \cdot (T_{\text{c}} + 5T_p) = 68.33$ seconds. In Figure 6a, we compare the convergence time of TED with those of the benchmark schemes through the cumulative density function (CDF) of ρ_{time} for a set of 1000 random topologies for three different $\text{P}_{\text{error,max}}$ values. The comparison is also made with two versions of TT-TDMA: one where $y = 4$ nodes are scheduled to share the time slot, and one where only $y = 2$ nodes transmit in the same time slot. The difference lies in the number of collisions allowed. The results show that compared to TDMA, and the two versions of TT-TDMA, the convergence time of TED is much shorter. While the results of Slotted Aloha are the worst, those of Aloha are better than TDMA and TT-TDMA. That is because the time slot in Slotted Aloha is close to the maximum propagation delay, as in TDMA, but the protocol is not collision-free. Since the pre-determined structure of TED is planned to optimize the number of link tests and since we consider convergence based on link test, as shown in Figure 5 the performance of TED also exceeds that of Aloha. We also observe that ρ_{time} decreases as $\text{P}_{\text{error,max}}$ increases. This is because for a higher packet error rate the topology is more sparse, and hence a node has to perform fewer link tests. However, the results of TED are almost independent of the packet error rate, which shows the robustness of our method. To comment on the effect of the assumed T_{c}, in Figure 6b we show the convergence time for different values of T_{c}. Note that while the operation of Aloha does not directly depend on the assumed value of T_{c}, link tests are measured for time windows of duration $2T_{\text{c}}$, and, as a result, ρ_{time} of ALOHA actually does change with T_{c}. However, this change is small since, while for a long link test window there is a higher chance to find non-colliding packets, at the same time the elapsed time for link test increases. Therefore, in Figure 6b we show ρ_{time} for ALOHA with $T_{\text{c}} = 1$ s, which was found to yield the best performance. We observe that TED exceeds the performance of all benchmark methods for $T_{\text{c}} > 1$ s (we note that TT-TDMA is affected similarly to TDMA). That is, TED is mostly affected when the distance between the nodes is known to be large (1500 m or beyond) or when it is unknown and thus assumed equal to the transmission range of the modems used. While the improved results of TED as T_{c} increases relative to those of Aloha seem to contradict the results in Fig. 5, we note that a large T_{c} trades off fewer packet collisions but longer transmission times and thus longer convergence time for



(a) Number of link tests

(b) Total number of received packets

Fig. 7: Per-node and per-link number of link tests (ρ_{link} from (2)) and packet receptions (ρ_{pck} from (3)) during the TDMA convergence time of $4N \cdot (T_{\text{c}} + 5T_p)$ seconds. Vertical lines show the 5% and 95% ranges. Results shown for $N = 8$, $L = 4$, and $T_p = 50$ ms, $T_{\text{c}} = 1.88$, $\text{P}_{\text{error,max}} = 0.05$. Results show that TED obtains a much higher number of link tests and received packets compared to TDMA and TT-TDMA, in the same time duration

topology discovery.

In Figure 7a, we compare the performance of TED in terms of the per-node and per-link number of link tests, ρ_{link} from (2). The results are compared with TT-TDMA, Aloha, Slotted Aloha, and with an *ideal TDMA* where all packets are assumed to be correctly received. For a fair comparison with TDMA for which $\rho_{\text{link}} = L$, we count packets until the end of the convergence time of TDMA, namely, for $4N \cdot (T_{\text{c}} + 5T_p)$ seconds. During this time, TDMA performs 4 link tests. In terms of number of link tests, the results show that TT-TDMA with $y = 2$ achieves the best results from all benchmark methods. This is because on one hand it better controls packet collisions than Aloha, Slotted Aloha, and TT-TDMA with $y = 4$, and on the other hand it allows more transmissions than TDMA. However, due to the controlled packet collisions and the large number of transmitted messages, the number of link tests performed by TED is significantly higher compared to all benchmark methods. Naturally, having a large number of link tests contributes to the accuracy in estimating the link reliability. Similar results are obtained in Figure 7b, where we show the per-node and per-link number of received packets, ρ_{pck} from (3), of TED, Aloha, Slotted Aloha, and TT-TDMA collected during the convergence time of TDMA. During this time, using TDMA ideally only $4 \cdot 5$ packets are received. Due to the large backoff time chosen for Aloha because of its best performance in terms of convergence time, here we observe that ρ_{link} and ρ_{pck} of Aloha are the same and in fact are less than the same metrics for ideal TDMA. However, the results show that even for high packet error rate much higher numbers are obtained using TED. The benefit of TED is observed also compared to TT-TDMA.

B. Results for Joint Link Discovery and Detection of NFNPs

In TED, the solution for joint link discovery and detection of NFNPs includes a single time frame of s time slots, each of $12T_{\mathrm{c}}$ seconds. Using the procedure discussed in Section IV-C, for $N = \{8, 16, 20\}$ nodes we obtain $s = \{16, 16, 25\}$ seconds, respectively. To determine a NFNP, we test if packets in one link are received with error rate below 0.1 while another link to the same receiver showed packet error rates above 0.1 but only due to the interference from the former. Measure ρ_{NFNP} from (4) is shown in Figure 8 as a function of N for the TED, TDMA, and TT-TDMA methods, where for fairness we match the total number of packet transmissions. As expected, since the number of time slots, s, increases with N, so does the number of detected NFNPs. Comparing the results of the three methods, we observe that the performance of the TT-TDMA method is poor. This is because, although TT-TDMA introduces spatial reuse, without proper scheduling and delays in transmissions, many packets still collide. In fact, due to the many collisions, the best results of TT-TDMA are obtained for $N = 8$. We also observe that the performance of TED is significantly better than that of TDMA. Since spatial reuse improves with N, this performance gain increases with N. In Figure 8, we also show the 5% and the 95% performance range for TED. We note that for $N > 8$ TED outperforms TDMA even in the lower 5% range.

Next, we show results for the detection of NFNPs. In Figure 9, we show the complementary cumulative density function (C-CDF) of ρ_{NFNP} from (4) for different $\mathrm{P}_{\mathrm{error,max}}$ levels (originated from different SL levels). We also show the C-CDF of the ground truth. We observe that even

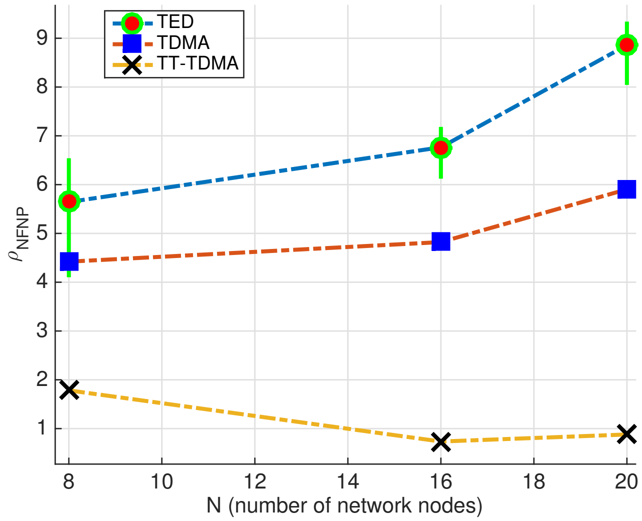


Fig. 8: Number of detected NFNPs, ρ_{NFNP} from (4), as a function of N. Vertical lines show the 5% and 95% ranges. Compared to TDMA and TT-TDMA, results show that using TED a significantly higher number of NFNPs can be detected.

for a high packet error rate of 5%, the number of NFNPs detected is not far from the optimal. However, for very high $\mathrm{P}_{\mathrm{error,max}}$ values beyond 10%, the performance significantly degrades. We also report that no erroneous NFNP was detected in the simulations. The results show that while the number of NFNPs varies with the topology, it is always significant. This motivates the use of information about NFNPs for network scheduling.

VI. Sea Experiment

While the packet error rate (PER) could be simulated using existing models for the time-varying underwater acoustic channel impulse response, our simulations do not include an evaluation of the link reliability as defined in Section III-C. This is because our definition of reliability requires the calculation of the probability to obtain a certain PER, whereas the simulated performance of TED is evaluated for a single channel realization. To fill this gap, in this section we complement the simulations by presenting results from multiple sea environments to investigate TED's ability to correctly detect communication links and evaluate their reliability.

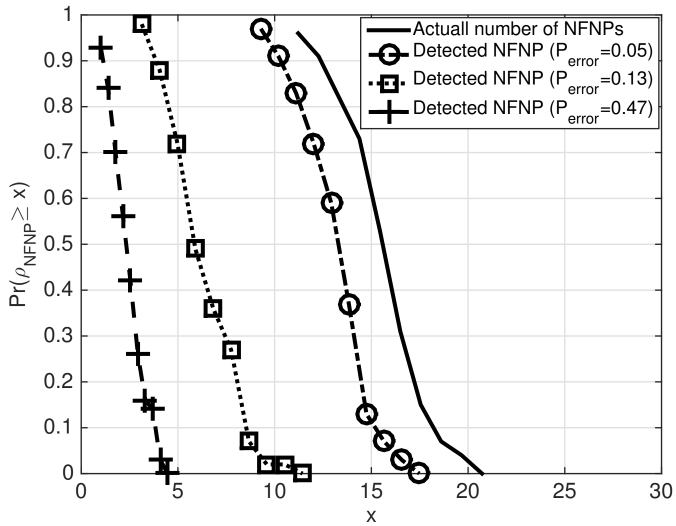
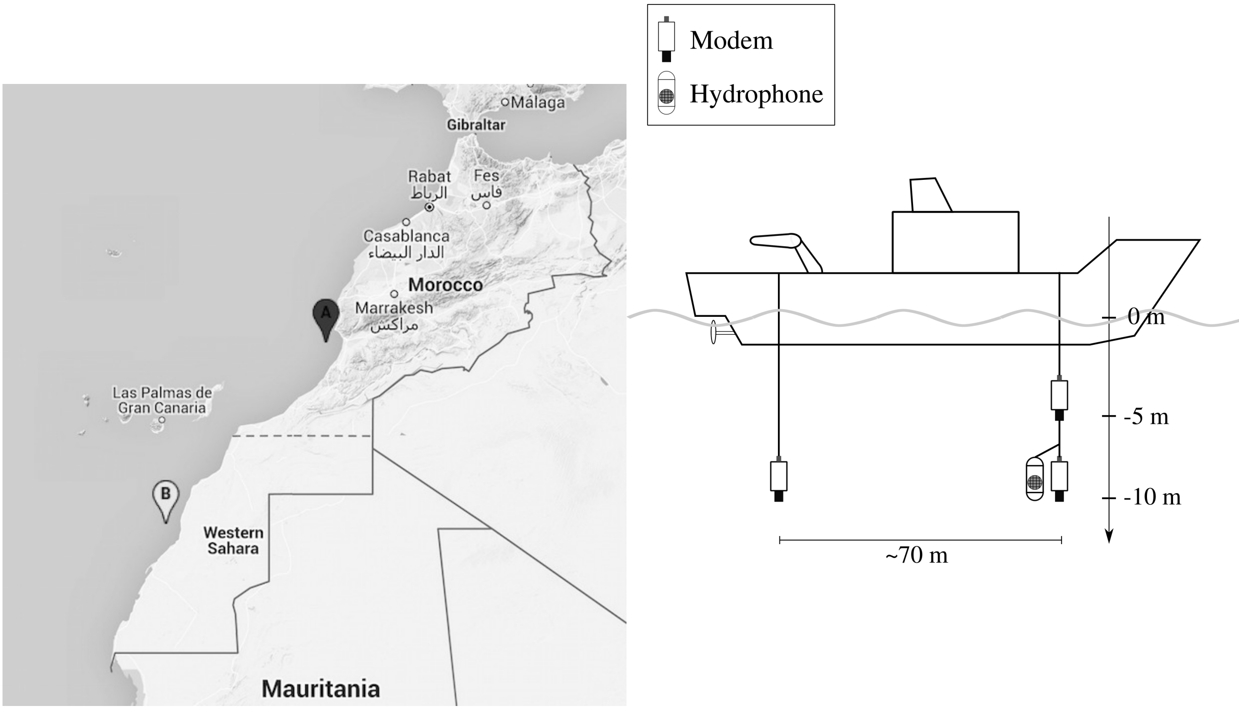


Fig. 9: C-CDF of the per-node number of detections of NFNPs, ρ_{NFNP} from (4), of TED. For a high packet error rate of 5%, misdetection is close to zero. Results show a high dependency of correct detections of NFNPs on the packet error rate.

A. Experiment Setup

The sea experiment was performed as part of the ALOMEX'15 expedition, led by the NATO CMRE institute, La Spezia, Italy. The experiment was performed onboard the 93 m long Alliance NATO research ship. During the experiment, TED was tested in two locations: off the coast of Morocco (water depth 1100 m) and off the coast of Western Sahara (water depth 50-80 m). These locations are shown in Figure 10a. In each location, we performed measurements in the morning and in the afternoon. Each test involved the deployment of three EvoLogics acoustic modems from the bow (at depth 5 m and 10 m) and the stern (at depth 10 m) of the Alliance. In addition, signals were recorded using a hydrophone unit deployed from the bow of the Alliance (at depth 10 m). The 10 m depth modem at the bow served as the receiver. In Figure 10b, we show an illustration of this setup.

The experiment tested the process of link discovery of TED (see Section IV-B). TED was executed using the NS2-based DESERT emulator [40] and a designated realtime synchronized scheduler connecting the application layer to the acoustic modems. Each transmission included $I = 10$ packets of 16 Bytes with total duration of $T_p = 100$ ms. The signals were transmitted



(a) (b)

Fig. 10: Setup of sea experiment: (a) Locations where TED was tested during the ALOMEX'15 experiment, (b) Illustration of deployment setup.

with carrier frequency of 20 kHz and a source level power of 175 dB re 1μPa at 1 m. In each location, TED was tested with different delay values, Δ_1 and Δ_2. During the experiment, to explore the impact of the modem used, we set Δ_1 and Δ_2 from (11) using different values of T_{c}. For the latter, while due to the very short propagation delay between the nodes in the experiment results would improve if we used low values for T_{c}, we test performance considering that the propagation delay is unknown and is thus set for the transmission range of the modems. Hence, as a best practice one would use the maximum propagation delay, i.e., the transmission range of the modem. In our case, we used values corresponding to the transmission range of current underwater modems.

B. Experimental Results

We require $L = 4$ packets properly received in order to detect a communication link. Analysis is made based on the logs of the modems and the acoustic raw data from the hydrophone. The

TABLE I. Results from the Sea Experiment. Rows show results for different test sites and times, and for different values of calculated $\mathbf{\Delta}$ based on different assumed T_c. Results are compared with theoretical ideal performance of TDMA and ALOHA, and with simulated ALOHA.

Location and Experiment duration [s]	$\mathbf{\Delta}$ [s]	**Assumed** T_c [s]	**TED**				**TDMA**		**ALOHA**	
			link ρ_pck	link ρ_time[s]	link ρ_link	$\mathrm{P}_\mathrm{reliab}$	Ideal ρ_pck	Ideal ρ_time[s]	Simulated ρ_time[s]	Ideal ρ_time[s]
A (afternoon) 383	[0.9, 1]	3	83.5	4.52	58.5	0.435	97	12.4	30.25	8.42
B (morning) 366	[0.6, 1]	4	260	4.7	150	0.81	73	16.4	27.58	8.42
B (noon) 313	[1.1, 1]	2.5	312	4.89	120.5	1	115	10.4	31.30	8.42
B (afternoon) 303	[1.2,0.5]	4.5	215.5	5.69	113.5	0.8	65	18.4	26.67	8.42
Average			**219**	**4.68**	**119.5**		**87**	**14.4**	**28.95**	**8.42**

former is used to calculate the average convergence time of TED (ρ_time from (1)), the average number of per-node link tests (ρ_link from (2)), and the average number of packets received in a given time of 300 s (ρ_pck from (3)). The received packets are used to measure the SINR of each successfully decoded packet, and based on (6), to calculate the link reliability $\mathrm{P}_\mathrm{reliab}$ for $P_\mathrm{error,req} = 10^{-5}$. Results are compared with the expected performance of a two-slot per time frame TDMA schedule with a time slot duration of $T_p + T_\mathrm{c}$, and considering an ideal scenario of error-free reception. To compare the results of TED with ALOHA, we conducted simulation results for the experimental setup and provide the average ρ_time for an assumed $\mathrm{P}_\mathrm{error,max} = 0.05$. We also give results for an *ideal* Aloha, where we assume a packet error rate of $\mathrm{P}_\mathrm{error,max} = 0.05$, 50% ideal offered load, throughput of $\epsilon = 15\%$, and we neglect the need to wait for link tests for convergence. Thus, for the fully connected network the ideal convergence time of ALOHA is calculated as $1/\left(1 - \mathrm{P}_\mathrm{error,max}\right) \cdot LN/\left(1/T_p \cdot \epsilon\right)$.

The results are summarized in Table I. As explained for the simulations, due to the tradeoff between the number of non-colliding packets found and the increase in the convergence time, we observe that ρ_{time} for the simulated ALOHA only slightly changes with T_{c}. We also observe the significant difference between the performance of the simulated ALOHA and that of the analyzed one. Clearly, this is because the latter does not consider the condition to test link stability. The results show that on average both in terms of the number of received packets and in terms of the number of link tests, results of TED are better than the performance of ideal TDMA. Although many collisions occurred, as reflected by the measured link reliability, we observe that the convergence time of TED is roughly one third of the anticipated convergence time of an ideal TDMA, half of the convergence time of ideal ALOHA, and much less than for the simulated ALOHA. Since for such a short range tested during the experiment the performance is mostly determined by the interference rate, we conclude that TED is insensitive to the tested communication link. Finally, since for each location TED was implemented with different Δ values for different assumed T_{c}, the results also verify the insensitivity of TED to different topology settings. This robustness is very important during the initial topology discovery phase of the network in real scenarios.

VII. Conclusions

In this paper, we considered the need to obtain reliable topology information in UWANs. Even for a small number of nodes, topology information is useful to determine destination nodes, perform routing, and schedule transmissions efficiently. With the aim to reduce the time overhead of the topology discovery phase while increasing the accuracy in evaluating the link's reliability, we proposed TED, the first designated topology discovery algorithm for UWANs. TED includes two efficient TDMA-based schedules: one designed only to detect reliable communication links, and the other to jointly detect links as well as scheduling conflicts in the form of near-far node pairs (NFNPs). In the former, communication links are discovered by ensuring that at least one packet per communication link is successfully received in each time frame. In the latter, packet collisions are deliberately enforced and NFNPs are detected by comparing the decoded packets with the expected ones. TED is effective for modems of large transmission range (around 1500 m or beyond) and works well for cases where the maximum propagation delay is large or assumed comparable to the transmission range of the modems used. Under these conditions, our numerical simulations showed that, compared to benchmark methods, TED obtains accurate topology information while greatly reducing the time overhead and increasing the accuracy in evaluating the link reliability. We also showed results from multiple sea environments collected during a 14 days sea experiment from southern Spain to the shores of west Africa. The results validated the numerical analysis and demonstrated the effectiveness of TED. Further work will be focused on how to utilize the discovery of NFNPs to improve network throughput, as well as obtaining experimental data with different propagation delays.

Acknowledgments

This work was supported by EKOE (Environmental Knowledge and Operation Effectiveness Program) from the NATO Centre for Maritime Research and Experimentation (CMRE). We would like to thank the captain and the crew of N.R.V. Alliance for their assistance during the data collection at sea. We would also like to thank collaborators from the ALOMEX'15 cruise who kindly provided access to ancillary datasets.

References

[1] K. Chen, M. Ma, E. Cheng, F. Yuan, and W. Su, "A survey on MAC protocols for underwater wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1433–1447, March 2014.

[2] R. Diamant, G. Shirazi, and L. Lampe, "Robust spatial reuse scheduling in underwater acoustic communication networks," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 32–46, January 2014.

[3] S. Gobriel, D. Mosse, and R. Cleric, "TDMA-ASAP: Sensor network TDMA scheduling with adaptive slot-stealing and parallelism," in *IEEE International Conference on Distributed Computing Systems (ICDCS'09)*, June 2009, pp. 458–465.

[4] A. Barnawi, "Adaptive TDMA slot assignment using request aggregation in wireless sensor networks," *Elsevier Procedia Computer Science*, vol. 10, pp. 78–85, December 2012.

[5] S. Shahabudeen, M. Motani, and M. Chitre, "Analysis of a high-performance MAC protocol for underwater acoustic networks," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 74–89, January 2014.

[6] W. Burdic, *Underwater Acoustic System Analysis*. Los Altos, CA, USA: Peninsula Publishing, 2002.

[7] M. Domingo, "Securing underwater wireless communication networks," *IEEE Wireless Communications*, vol. 18, no. 1, pp. 22–28, February 2011.

[8] A. Othman, W. Abidin, H. Zen, K. Ping, and N. Zamhari, "Performance evaluation for network set up and nodes discovery protocol in underwater grid topology networks," in *Information and Telecommunication Technologies (APSITT)*, June 2010, pp. 1–6.

[9] R. Diamant and L. Lampe, "Spatial reuse time-division multiple access for broadcast ad hoc underwater acoustic communication networks," *IEEE Journal of Oceanic Engineering*, vol. 36, no. 2, pp. 172–185, April 2011.

[10] Y. Noh, U. Lee, S. Han, P. Wang, D. Torres, J. Kim, and M. Gerla, "DOTS: A propagation delay-aware opportunistic MAC protocol for mobile underwater networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 4, pp. 766–782, April 2014.

[11] P. Casari and M. Zorzi, "Protocol design issues in underwater acoustic networks," *Elsevier Computer Communications*, vol. 34, no. 17, pp. 2013–2025, June 2011.

[12] R. Diamant, P. Casari, F. Campagnaro, and M. Zorzi, “A handshake-based protocol exploiting the near-far effect in underwater acoustic networks,” *IEEE Wireless Communications Letters*, vol. 5, no. 3, pp. 308–311, June 2016.

[13] A. Muqattash, M. Krunz, and W. Ryan, “Solving the near-far problem in CDMA-based ad hoc networks,” *Ad Hoc Networks*, vol. 1, no. 4, pp. 435–453, November 2003.

[14] D. Pompili and I. Akyildiz, “Overview of networking protocols for underwater wireless communications,” *IEEE Communications Magazine*, vol. 47, no. 1, pp. 97–102, January 2009.

[15] J. Zhang and Y. Zheng, “Frequency-domain turbo equalization with soft successive interference cancellation for single carrier MIMO underwater acoustic communications,” *IEEE Transactions on Wireless Communications*, vol. 10, no. 9, pp. 2872–2882, September 2011.

[16] A. Syed, W. Ye, B. Krishnamachari, and J. Heidemann, “Understanding spatio-temporal uncertainty in medium access with ALOHA protocols,” in *International Workshop on UnderWater Networks (WUWNet)*, Montréal, Canada, September 2007.

[17] J. Ahna, A. Syedb, B. Krishnamachari, and J. Heidemann, “Design and analysis of a propagation delay tolerant ALOHA protocol for underwater networks,” *Elsevier Ad Hoc Networks Journal*, vol. 9, no. 5, pp. 752–766, Jul. 2011.

[18] J. Heidemann, M. Stojanovic, and M. Zorzi, “Underwater sensor networks: applications, advances and challenges,” *Phil. Trans. R. Soc. A*, vol. 370, no. 1958, pp. 158–175, 2012.

[19] M. Molins and M. Stojanovic, “Slotted FAMA: a MAC protocol for underwater acoustic networks,” in *IEEE OCEANS Conference*, Singapore, May 2006.

[20] I. Rhee, A. Warrier, M. Aia, J. Min, and M. L. Sichitiu, “Z-MAC: a hybrid MAC for wireless sensor networks,” *IEEE/ACM Transactions on Networking (TON)*, vol. 16, no. 3, pp. 511–524, June 2008.

[21] B. Zeng, Y. Dong, and Z. Liu, “A negotiation-based TDMA scheduling with consecutive slots assignment for wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 2014, August 2014.

[22] Z. Cai, M. Lu, and C. Georghiades, “Topology-transparent time division multiple access broadcast scheduling in multihop packet radio networks,” *IEEE Transactions on Vehicular Technology*, vol. 52, no. 4, pp. 970–984, July 2003.

[23] Q. Sun, V. Li, and K. Leung, “A framework for topology-transparent scheduling in wireless networks,” in *IEEE Vehicular Technology Conference (VTC)*, May 2010, pp. 1–5.

[24] ——, “Adaptive topology-transparent distributed scheduling in wireless networks,” in *IEEE International Conference on Communications (ICC)*, May 2010, pp. 1–5.

[25] Y. Liu, V. Li, K. Leung, and L. Zhang, “Performance improvement of topology-transparent broadcast scheduling in mobile ad hoc networks,” *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4594–4605, November 2014.

[26] D. Son, B. Krishnamachari, and J. Heidemann, “Experimental study of the effects of transmission power control and blacklisting in wireless sensor networks,” in *IEEE Sensor and Ad Hoc Communications and Networks (SECON)*, October 2004, pp. 289–298.

[27] J. Xu, K. Li, and G. Min, “Reliable and energy-efficient multipath communications in underwater sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 7, pp. 1326–1335, July 2012.

[28] A. Nimbalkar and D. Pompili, “Reliability in underwater inter-vehicle communications,” in *Proceedings of the third ACM international workshop on Underwater Networks*, September 2008, pp. 19–26.

[29] M. A. Luque-Nieto, J. M. Moreno-Roldan, J. Poncela, and P. Otero, “Reliable transmission in fair STDMA underwater networks,” in *Computing for Sustainable Global Development (INDIACom)*, March 2015.

[30] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 84–89, January 2009.

[31] J. Rice, B. Creber, C. Fletcher, P. Baxley, K. Rogers, K. McDonald, D. Rees, M. Wolf, S. Merriam, R. Mehio *et al.*, "Evolution of seaweb underwater acoustic networking," in *MTS/IEEE Oceans Conference*, vol. 3, February 2000, pp. 2007–2017.

[32] A. Syed and J. Heidemann, "Time synchronization for high latency acoustic networks," in *IEEE Conference on Computer Communications (INFOCOM)*, Barcelona, Spain, April 2006.

[33] R. Diamant and L. Lampe, "Underwater localization with time-synchronization and propagation speed uncertainties," *IEEE Transactions on Mobile Computing*, vol. 12, no. 7, pp. 1257–1269, July 2013.

[34] F. Blasco, F. Rossetto, and G. Bauch, "Time interference alignment via delay offset for long delay networks," *IEEE Transactions on Communications*, vol. 62, no. 2, pp. 590–599, 2014.

[35] R. Diamant, W. Shi, W. Soh, and L. Lampe, "Joint time and spatial reuse handshake protocol for underwater acoustic communication networks," *IEEE J. Oceanic Eng.*, vol. 38, no. 3, pp. 470–483, Jul. 2013.

[36] R. Diamant, Y. Bucris, and A. Feuer, "A reliability measure and reliability evaluation of underwater acoustic communication," *Elsevier journal of ocean engineering and science*, no. 2, April 2016.

[37] J. Bezdek and R. Hathaway, "Some notes on alternating optimization," in *International Conference on Fuzzy Systems (AFSS)*, London, UK, February 2002, pp. 288–300.

[38] M. Porter and H. Bucker, "Gaussian beam tracing for computing ocean acoustic fields," *Journal of Acoustical Society of America*, vol. 82, no. 4, pp. 1349–1359, October 1987.

[39] N. Abramson, "The ALOHA system," *Proceedings of AFIPS Fall Joint Computer Conference*, vol. 37, pp. 281–285, 1970.

[40] P. Casari, C. Tapparello, F. Guerra, F. Favaro, I. Calabrese, G. Toso, S. Azad, R. Masiero, and M. Zorzi, "Open source suites for underwater networking: WOSS and DESERT underwater," *IEEE Network*, vol. 28, no. 5, pp. 38–46, September 2014.