



Deposited via The University of York.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/id/eprint/174608/>

Version: Published Version

Article:

Morozs, Nils, Mitchell, Paul Daniel and Zakharov, Yury (2021) LTDA-MAC v2.0: Topology-Aware Unsynchronized Scheduling in Linear Multi-Hop UWA Networks. *Network*. pp. 2-10. ISSN: 2673-8732

<https://doi.org/10.3390/network1010002>

Reuse

This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here:

<https://creativecommons.org/licenses/>

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

Article

LTDA-MAC v2.0: Topology-Aware Unsynchronized Scheduling in Linear Multi-Hop UWA Networks [†]

Nils Morozs ^{*} , Paul D. Mitchell  and Yuriy Zakharov 

Department of Electronic Engineering, University of York, York YO10 5DD, UK;
paul.mitchell@york.ac.uk (P.D.M.); yury.zakharov@york.ac.uk (Y.Z.)

^{*} Correspondence: nils.morozs@york.ac.uk

[†] This work was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through the USMART (EP/P017975/1) and Full-Duplex (EP/R003297/1) projects. This research was performed on the Viking Cluster—a high performance computing (HPC) facility provided by the University of York.

Abstract: This paper investigates the use of underwater acoustic sensor networks (UASNs) for subsea asset monitoring. In particular, we focus on the use cases involving the deployment of networks with line topologies, e.g., for monitoring oil and gas pipelines. The Linear Transmit Delay Allocation MAC (LTDA-MAC) protocol facilitates efficient packet scheduling in linear UASNs without clock synchronization at the sensor nodes. It is based on the real-time optimization of a packet schedule for a given network deployment. In this paper, we present a novel greedy algorithm for real-time optimization of LTDA-MAC schedules. It produces collision-free schedules with significantly shorter frame duration, and is 2–3 orders of magnitude more computationally efficient than our previously proposed solution. Simulations of a subsea pipeline monitoring scenario show that, despite no clock synchronization, LTDA-MAC equipped with the proposed schedule optimization algorithm significantly outperforms Spatial TDMA.



Citation: Morozs, N.; Mitchell, P.D.; Zakharov, Y. LTDA-MAC v2.0: Topology-Aware Unsynchronized Scheduling in Linear Multi-Hop UWA Networks. *Network* **2021**, *1*, 2–10. <https://doi.org/10.3390/network1010002>

Academic Editor: Safdar H. Bouk

Received: 30 April 2021

Accepted: 21 May 2021

Published: 25 May 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: linear TDA-MAC; Medium Access Control; underwater acoustic network

1. Introduction

Underwater acoustic sensor networks (UASNs) are a key technology for monitoring the underwater environment, enabled by the recent developments in underwater acoustic modem technologies [1–5]. Acoustic waves are the preferred communication medium underwater, since they can propagate significantly further than electromagnetic and optical waves. However, acoustic communications are fundamentally limited by the slow sound propagation and low available bandwidth. These physical constraints present the need for Medium Access Control (MAC) protocols designed specifically for UASNs.

A major application area of UASNs is live monitoring of subsea oil and gas infrastructure, which may often involve sensor networks with line topologies, e.g., for leakage detection and corrosion monitoring in underwater pipelines [6,7]. In linear UASNs (LUASNs) the packets are routed via multiple hops between neighbouring nodes as shown in Figure 1. Typical connection ranges in LUASNs allow the nodes to communicate with their neighbours, but are short enough to avoid interference between nodes further apart. Such sparse node connectivity is highly suitable for spatial spectrum reuse, i.e., simultaneous packet scheduling in several parts of the network without collisions.

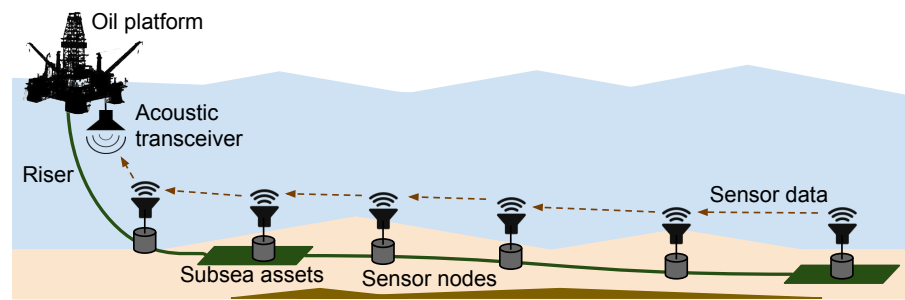


Figure 1. Example of a linear UASN deployment in a subsea asset monitoring scenario (Reprinted with permission from ref. [8]. Copyright 2019 IEEE Networking Letters.).

The state-of-the-art research on MAC protocols for UASNs with sparsely connected topologies (e.g., LUASNs) focuses on designing spatial reuse patterns of slots in TDMA schedules [9–11]. A significant drawback of TDMA-based protocols is the need for long guard intervals to account for the propagation delays, thus reducing the network throughput and increasing the packet latency [12]. They also require clock synchronization among the network nodes, which is more challenging in UASNs than in terrestrial networks [13,14]. As an alternative to deterministic schedule-based TDMA methods, communication networks often use contention-based MAC protocols [15–19] where nodes access a shared channel randomly on demand, based on a particular set of rules [20]. However, most conventional contention-based MAC protocols are highly inefficient in the underwater acoustic environment. For example, channel reservation based protocols waste a large part of channel capacity while the nodes are waiting for control signals to propagate through the slow acoustic medium to establish a communication link, e.g., Request-to-Send (RTS), Clear-to-Send (CTS), acknowledgements etc. These waiting times result in significant loss of throughput and poor channel utilization [13,15,16]. This paper focuses on the schedule-based MAC approach, as it is particularly well-suited for the linear network topologies, where the sparse node connectivity can be exploited to schedule simultaneous, spatially separated transmissions.

In [8] we proposed the Linear Transmit Delay Allocation MAC (LTDA-MAC) protocol that achieves high throughput in LUASNs via heuristic optimization of packet schedules that do not require clock synchronization at the network nodes. It works for arbitrary node connectivity/interference patterns, exploiting long propagation delays and topology sparsity specific to a given network deployment. The purpose of this paper is to report on a significant further development on LTDA-MAC: a novel greedy optimization algorithm that produces better scheduling solutions and dramatically reduces the computational cost compared with our previous method based on global optimization via a Genetic Algorithm (GA) and Particle Swarm Optimization (PSO).

The rest of the paper is organized as follows: Section 2 presents the details of LTDA-MAC and our proposed schedule optimization algorithm, Section 3 discusses the simulation results, and Section 4 concludes the paper.

2. Linear TDA-MAC

The TDA-MAC protocol [21] was originally designed to facilitate centralized packet scheduling in single-hop UASNs, i.e., where multiple sensor nodes are connected to a single master node. It was then extended to support dual-hop topologies in [22] and line network topologies in [8]. Its key differentiating feature compared with other schedule-based MAC protocols is that it does not require clock synchronization at the sensor nodes. Instead, the times of transmissions are determined locally at the nodes, based on the node-specific delays between receiving a request packet and transmitting their data packets.

In this paper we focus on unsynchronized scheduling in LUASNs using the LTDA-MAC protocol [8]—a protocol designed specifically for networks with line topologies. Figure 1 depicts an example of a LUASN. Every sensor node only uses two connections—a

node one hop closer to the sink node (*up the chain*) and a node one hop further *down the chain*. The job of a sensor node is to transmit its own packets up the chain and forward data packets from sensor nodes down the chain.

Figure 2 depicts an LTDA-MAC schedule, in this particular example based on one-hop interference range among the sensor nodes. In addition to scheduling their own data transmissions, Nodes 1 and 2 are responsible for forwarding the packets between the sink node and the other nodes down the chain. The LTDA-MAC schedule is defined by the delays between sensor nodes receiving a request (REQ) packet and transmitting a data packet. For a network comprising N_{sn} sensor nodes, the transmit delays that define an LTDA-MAC schedule can be expressed by a triangular matrix:

$$\mathbf{T}_{tx} = \begin{bmatrix} T_{tx}[1,1] & T_{tx}[1,2] & \cdots & T_{tx}[1,N_{sn}] \\ \emptyset & T_{tx}[2,2] & \cdots & T_{tx}[2,N_{sn}] \\ \vdots & \vdots & \ddots & \vdots \\ \emptyset & \emptyset & \cdots & T_{tx}[N_{sn},N_{sn}] \end{bmatrix}, \quad (1)$$

where $T_{tx}[i,i]$ is the transmit delay assigned to node i for sending its own data packets, $T_{tx}[i,j]$ ($i < j$) is the transmit delay assigned to node i for forwarding data packets originating at node j .

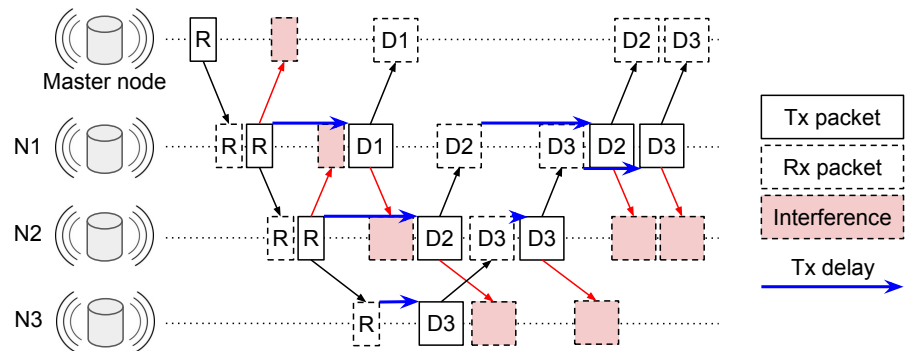


Figure 2. Example of an LTDA-MAC schedule, where the master node gathers the data from three sensor nodes. R—REQ packet, D—data packet (Reprinted with permission from ref. [8]. Copyright 2019 IEEE Networking Letters).

2.1. LTDA-MAC Schedule Optimization

Deriving an LTDA-MAC schedule for a given LUASN deployment requires a joint optimization of $N_{sn}(N_{sn} + 1)/2$ values in \mathbf{T}_{tx} that minimize the frame duration:

$$\min \tau_{\text{frame}}(\mathcal{N}, \mathbf{T}_{tx}) \quad (2a)$$

$$\text{s.t. } n_{\text{col}}(\mathcal{N}, \mathbf{T}_{tx}, \tau_g) = 0, \quad (2b)$$

where $\tau_{\text{frame}}(\mathcal{N}, \mathbf{T}_{tx})$ is defined as the time between the sink node sending the initial REQ packet and receiving the final data packet, $n_{\text{col}}(\mathcal{N}, \mathbf{T}_{tx}, \tau_g)$ is the number of collisions caused by the transmitted/received packets overlapping in time at any node, τ_g is the *desired guard interval* specifying the minimum allowed time separation between scheduled packets, and \mathcal{N} is a representation of a given network topology described below.

We define the network deployment parameters as a tuple $\mathcal{N} = \{I, \mathbf{T}_p, \tau_{rp}, \tau_{dp}\}$, where I is the interference matrix, \mathbf{T}_p is the matrix of propagation delays between every pair of nodes, and τ_{rp} and τ_{dp} are the REQ and data packet duration, respectively. I and \mathbf{T}_p are computed by the master node during the network discovery stage, and can then be periodically updated based on the timing of received data packets. This process is appropriate for tracking the topology of a quasi-stationary UASN, as described in [21]. I and \mathbf{T}_p are $N \times N$ matrices, where $N = 1 + N_{sn}$ is the total number of nodes, including one master (sink) node and N_{sn} sensor nodes. I is a binary matrix, where $I[i,j] = 1$ if node j

can interfere with node i , and $I[i, j] = 0$ otherwise. $T_p[i, j]$ is the propagation delay from node i to node j .

The number of packet collisions in a candidate LTDA-MAC schedule $n_{\text{col}}(\mathcal{N}, T_{\text{tx}}, \tau_g)$ depends on the specific pattern of interfering links, propagation delays and packet durations, different for every network deployment. We found it intractable to solve this optimization problem analytically for an arbitrary network topology \mathcal{N} (a formal proof of complexity of this problem is a subject of further work). Therefore, in [8] we proposed a heuristic “simulation-in-the-loop” optimization algorithm comprising two stages: (1) GA for initial coarse optimization, (2) PSO for converging on a locally optimal solution. There, the packet collision constraint (2b) is evaluated algorithmically by calculating the Tx and Rx times of every packet in a single frame. If any pair of Tx/Rx packets overlaps in time at the same node, or is separated by less than τ_g , a collision is detected and $n_{\text{col}}(\mathcal{N}, T_{\text{tx}}, \tau_g)$ is incremented. The drawback of this approach is its need to evaluate many candidate solutions before converging. The purpose of this paper is to propose an alternative heuristic optimization algorithm that produces better LTDA-MAC schedule solutions and that is significantly more computationally efficient.

2.2. Greedy Optimization Algorithm

The high computational complexity of the GA + PSO method proposed in [8] stems from the need to minimize a nonlinear discontinuous objective function in $N_{\text{sn}}(N_{\text{sn}} + 1)/2$ dimensions, resulting in a vast non-convex solution space. To reduce the computational cost of LTDA-MAC schedule optimization we propose a new greedy algorithm that iterates over every Tx delay in T_{tx} and chooses a locally optimal value for it in isolation. To ensure that valid LTDA-MAC schedules are produced by such an iterative greedy optimization approach, we impose the following constraints on the Tx delays.

Firstly, the minimum transmit delay assigned to node n for transmitting its own data packet is:

$$\forall n \in \{1..N_{\text{sn}}\}, T_m[n, n] = \begin{cases} \tau_{\text{rp}} + 2\tau_g, & n < N_{\text{sn}} \\ \tau_g, & n = N_{\text{sn}}, \end{cases} \quad (3)$$

where $T_m[i, j]$ is the minimum transmit delay that can be assigned to node i for transmitting a packet generated by node j . The above constraint ensures that those nodes that need to forward a REQ packet further down the chain, have time to do it before transmitting their data packet up the chain.

Secondly, the minimum transmit delay for forwarding other nodes' packets is defined as follows:

$$\forall n, k \in \{1..N_{\text{sn}}\}, k > n, \quad (4)$$

$$T_m[n, k] = 2(\tau_p[n+1] + \tau_g) + \tau_{\text{rp}} + \tau_{\text{dp}} + T_{\text{tx}}[n+1, k],$$

where $\tau_p[i]$ is the propagation delay on the i^{th} link between adjacent nodes of the linear network topology, i.e., $\forall i \in \{1..N_{\text{sn}}\}, \tau_p[i] = T_p[i, i+1]$.

The above constraint ensures that a node cannot forward a data packet from another node before actually receiving it. Note that in this paper all values in T_{tx} are relative to the time of the REQ packet reception, i.e., $T_{\text{tx}}[i, j]$ is defined as the time between node i receiving a REQ packet and transmitting the data packet generated by node j . This is different to the original definition of T_{tx} in [8]. This new definition decouples the Tx delays assigned to multiple nodes for forwarding the same data packet up the chain, and enables our proposed greedy algorithm to optimize them in isolation.

Our proposed greedy optimization method is shown in Algorithm 1. It iterates over every value in T_{tx} in the order that maximizes the chances of scheduling simultaneous spatially separated transmissions. For every Tx delay $T_{\text{tx}}[i, j]$, the algorithm starts by evaluating the LTDA-MAC schedule using $T_{\text{tx}}[i, j] = T_m[i, j]$, i.e., the minimum possible value according to (3) and (4). If there are collisions in the schedule, $T_{\text{tx}}[i, j]$ is incremented by τ_{step} , and the schedule is evaluated again. This incremental search continues until the schedule is collision-free. In the experiments reported in this paper the value of τ_{step} was

set to $\tau_{\text{guard}}/2$ which achieved a good trade-off between the algorithm's ability to find good solutions and its computational cost. Incorporating more sophisticated variable step methods is a subject of further work.

Algorithm 1 The proposed greedy optimization algorithm for LTDA-MAC scheduling

```

1: Create  $\mathcal{N}$  via initial network discovery
2: Set the desired guard interval  $\tau_g$  and time step  $\tau_{\text{step}}$ 
3: Initialize collision-free schedule  $T_{\text{tx}}$  using (5)
4: for  $i \in \{1..N_{\text{sn}}\}$  do
5:   for  $n \in \{1..(N_{\text{sn}} - i + 1)\}$  do
6:     Calculate the packet index  $k = n + i - 1$ 
7:     Calculate  $T_{\text{m}}[n, k]$  using (3) if  $n = k$ , or (4) if  $n \neq k$ 
8:     Initialize Tx delay:  $T_{\text{tx}}[n, k] = T_{\text{m}}[n, k]$ 
9:     while  $n_{\text{col}}(\mathcal{N}, T_{\text{tx}}, \tau_g) > 0$  do
10:      Increment Tx delay:  $T_{\text{tx}}[n, k] \leftarrow T_{\text{tx}}[n, k] + \tau_{\text{step}}$ 
11:    end while
12:   end for
13: end for

```

The Tx delay matrix is initialized as follows:

$$\forall n, k \in \{1..N_{\text{sn}}\}, k \geq n, T_{\text{tx}}[n, k] = (N_{\text{sn}}n + k) T_{\text{large}}, \quad (5)$$

where T_{large} is an arbitrarily long time interval, e.g., we use $T_{\text{large}} = 10^6$, that makes sure that the initial Tx delay matrix is collision-free. It also allows every Tx delay in T_{tx} to be optimized separately and replace its initial arbitrary value without affecting the rest of the schedule.

3. Simulation Results

3.1. Simulation Setup

This section evaluates the performance of the LTDA-MAC protocol equipped with the proposed schedule optimization algorithm applied to the pipeline monitoring scenarios considered in [8], similar to that depicted in Figure 1. The maximum sea depth is 500 m. The pipeline at 480 m depth is connected to the platform at the sea surface through a riser, whose shape is modeled as a quarter-circle. Two different pipeline lengths are simulated: 2 km and 20 km. Initially, 11 nodes (1 sink + 10 sensor nodes) are spread across the length of the pipeline at equidistant points. Afterwards, 50 sets of node positions are generated to include random horizontal perturbations (0–20 m and 0–200 m for the two scenarios, respectively) in random directions from the initial equidistant points. This approach yields a statistical spread of channel conditions with variations in received signal and interference power, thus providing statistically more valid results. The wideband multipath channel with 24 kHz centre frequency and 7.2 kHz bandwidth was modeled using the BELLHOP beam tracing method described in [23]. Without loss of generality, these simulations use a threshold for interfering link detection of 0 dB Signal-to-Noise Ratio (SNR), i.e., all links with $\text{SNR} \geq 0$ dB were marked with 1 in the interference matrix I , while all of the wanted links in the linear topology also have $\text{SNR} > 0$ dB. The SNR is calculated using the analytical ambient noise model with 10 m/s wind speed and 0.5 shipping activity factor [23]. The other parameters for the 2 km and 20 km pipeline scenarios, respectively, are:

1. 140 dB re $\mu\text{Pa}^2\text{m}^2$ source level, 200 ms data packets, 50 ms REQ packets, 25 ms guard interval;
2. 170 dB re $\mu\text{Pa}^2\text{m}^2$ source level, 500 ms data packets, 100 ms REQ packets, 100 ms guard interval.

3.2. Discussion

Figure 3 compares the LTDA-MAC schedules derived by the greedy algorithm proposed in this paper with those derived by our previously proposed GA + PSO method [8]. The parameters of the GA and PSO algorithms are:

- GA: population size—500, mutation rate—0.1, 80% scattered crossover, 1000 generations limit;
- PSO: swarm size—500, minimum neighbourhood fraction—0.1, adaptive inertia range—[0.05, 0.8], 1000 iterations limit.

These parameters were empirically found to produce the best performance in our previous study in [8], by providing a large enough population/swarm to explore the high-dimensional solution space, and to enable a sufficient number of GA generations and PSO iterations to find good suboptimal solutions.

This section also provides a baseline comparison with the Spatial TDMA approach (STDMA) [10,11], tailored to the linear network scenario studied in this paper. An STDMA frame comprises a sequence of synchronized time slots for collision-free multiple access. To avoid inter-slot interference, the slot duration is:

$$\tau_{\text{slot}} = \tau_{\text{dp}} + \max_{l[i,j]=1} \{T_p[i,j]\} + \tau_g, \quad (6)$$

which ensures that no packets transmitted in a given TDMA slot are received at any node in the subsequent slot. The spatial reuse pattern is achieved by computing a binary $N_{\text{sn}} \times L$ matrix indicating which node transmits in which time slot, such that L (the number of slots) is minimized with no collisions. The STDMA frame length is then calculated as: $\tau_{\text{frame}} = L\tau_{\text{slot}}$.

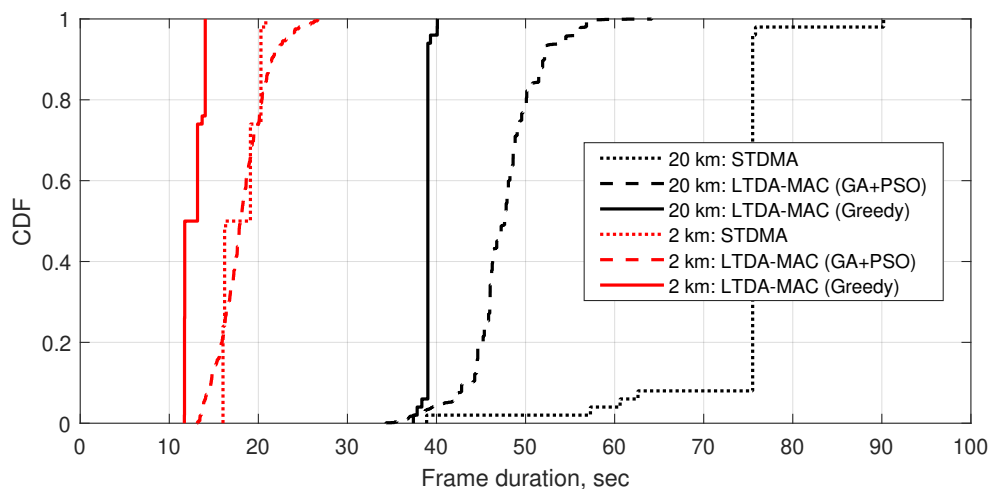


Figure 3. The proposed greedy optimization algorithm provides significantly better LTDA-MAC packet schedules (shorter frame duration—higher throughput), compared with the previously proposed “GA + PSO” method and the conventional Spatial TDMA approach.

Figure 3 shows that the new heuristic optimization algorithm proposed in this paper derives packet schedules with shorter frame duration than the GA + PSO method and the STDMA protocol (despite the latter benefiting from clock synchronization), thus achieving higher network throughput of the LTDA-MAC protocol in both simulated scenarios. Furthermore, the proposed greedy algorithm is more reliable than GA + PSO, because the former is a deterministic method that does not rely on randomly generating and mutating candidate solutions. Therefore, the statistical spread of the results achieved by the greedy algorithm is significantly smaller, and is caused only by the perturbations in the simulated network topologies, rather than the random number generator seed (like GA, PSO and similar methods). The significant difference in the optimization performance

between the proposed greedy algorithm and the GA + PSO method is due to the inherently discontinuous and high-dimensional nature of the optimization problem, where treating the entire scheduling problem as a “black box” makes it very challenging for a heuristic algorithm such as GA or PSO to find good solutions. In contrast, the new greedy algorithm deconstructs the problem into a series of much simpler problems, minimizing every transmit delay in isolation, thus resulting in much better performance. Figure 4 gives examples of the LTDA-MAC schedules derived for the 2 km and 20 km pipeline scenarios, showcasing the topology-aware spatial reuse of resources achieved by the LTDA-MAC protocol employing the proposed schedule optimization algorithm.

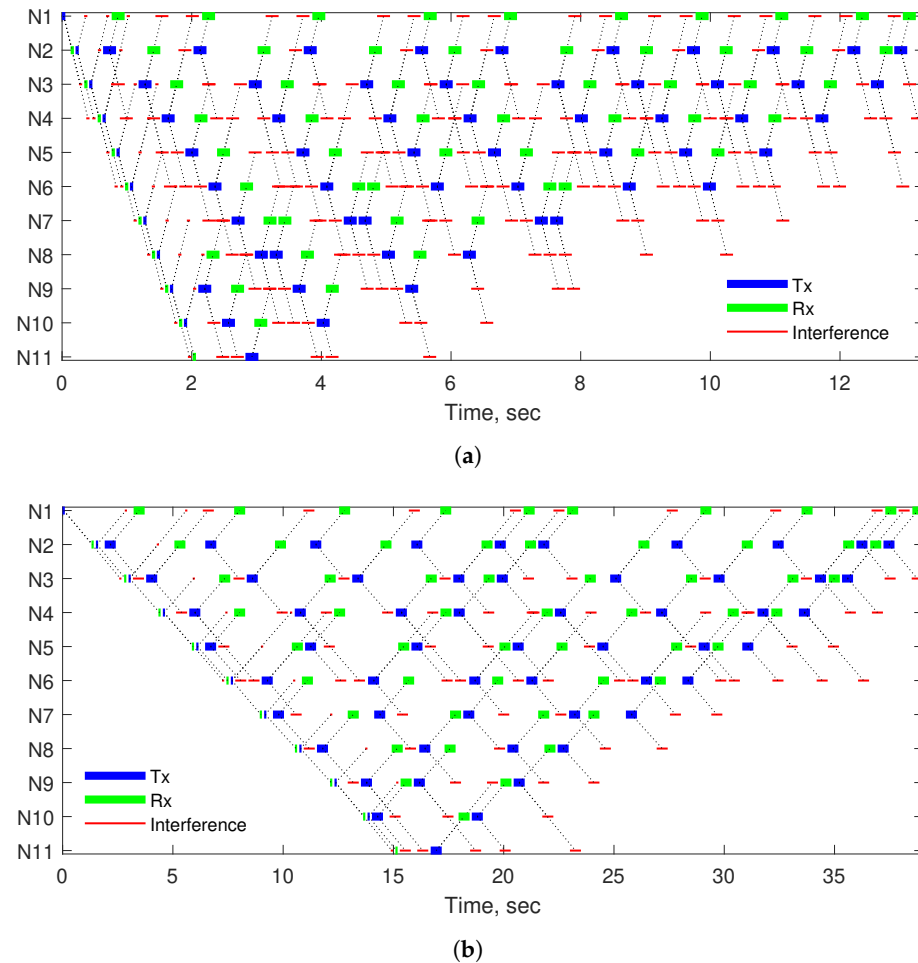


Figure 4. LTDA-MAC schedules derived by the proposed greedy algorithm for 11-node linear UASNs deployed on 2 km and 20 km long pipelines. (a) A 2 km long network (200 m average distance between adjacent nodes); (b) 20 km long network (2 km average distance between adjacent nodes).

In addition to the notable improvement in the algorithm performance shown in Figure 3, Table 1 highlights the significant reduction in the computational cost of the greedy optimization algorithm compared with our previously proposed GA + PSO method. The number of schedule evaluations before finding a solution is reduced by 2–3 orders of magnitude, thus reducing the computer runtime from several minutes to approximately 1 s or less. The runtime figures were obtained by executing the algorithms in MATLAB R2018a as single-core jobs on the standard node of the Viking computing cluster at University of York: Intel Xeon 6138 2.0 GHz, Centos Linux 7.3.

Table 1. Number of schedule evaluations (M—million) and computer runtime of the proposed greedy optimization algorithm, compared with the GA + PSO method.

Metric	Scenario	Algorithm Performance (Mean 5th–95th Percentile Range)	
		GA + PSO	Greedy
Num. evals	2 km pipeline	(0.99M 0.90M–1M)	3402 (2871–4212)
	20 km pipeline	0.99M (0.89M–1M)	821 (754–827)
Runtime, sec	2 km pipeline	424 (337–489)	1.1 (0.91–1.4)
	20 km pipeline	384 (312–454)	0.27 (0.25–0.34)

4. Conclusions

The LTDA-MAC protocol facilitates efficient packet scheduling in LUASNs without the need for synchronized clocks at the network nodes. The key part of the LTDA-MAC protocol is the real-time application of a heuristic optimization algorithm to produce a packet schedule with spatial spectrum reuse tailored to the given deployment scenario. In this paper, we proposed a new greedy optimization algorithm for deriving LTDA-MAC packet schedules that produces reliably better solutions and is 2–3 orders of magnitude more computationally efficient than our previous GA + PSO optimization method. Simulations of LUASNs deployed on 2 km and 20 km underwater pipelines showed that LTDA-MAC can exploit long propagation delays and the interference pattern in a given network deployment to considerably improve the network throughput, compared with the conventional STDMA approach, despite the lack of clock synchronization in LTDA-MAC. Continuing from our previous work on single-hop and dual-hop TDA-MAC, LTDA-MAC equipped with the new greedy schedule optimization algorithm is a key step in our further work on TDA-MAC for unsynchronized scheduling in UASNs with general multi-hop topologies.

Author Contributions: Conceptualization, N.M.; methodology, N.M., P.D.M. and Y.Z.; software, N.M.; validation, N.M., P.D.M. and Y.Z.; formal analysis, N.M.; investigation, N.M., P.D.M. and Y.Z.; resources, P.D.M. and Y.Z.; data curation, N.M.; writing—original draft preparation, N.M.; writing—review and editing, P.D.M., Y.Z.; visualization, N.M.; project administration, P.D.M.; funding acquisition, P.D.M., Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through the USMART (EP/P017975/1) and Full-Duplex (EP/R003297/1) projects.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Dol, H.S.; Casari, P.; van der Zwan, T.; Otnes, R. Software-Defined Underwater Acoustic Modems: Historical Review and the NILUS Approach. *IEEE J. Ocean. Eng.* **2017**, *42*, 722–737. [[CrossRef](#)]
- Demirors, E.; Sklivanitis, G.; Santagati, G.E.; Melodia, T.; Batalama, S.N. A High-Rate Software-Defined Underwater Acoustic Modem with Real-Time Adaptation Capabilities. *IEEE Access* **2018**, *6*, 18602–18615. [[CrossRef](#)]
- Renner, C.; Golkowski, A. Acoustic Modem for Micro AUVs: Design and Practical Evaluation. In Proceedings of the ACM WUWNet'16, Shanghai, China, 24–26 October 2016; pp. 2:1–2:8.
- Sherlock, B.; Tsimenidis, C.C.; Neasham, J.A. Signal and receiver design for low-power acoustic communications using m-ary orthogonal code keying. In Proceedings of the IEEE OCEANS 2015, Genova, Italy, 18–21 May 2015. [[CrossRef](#)]
- Zakharov, Y.; Henson, B.; Diamant, R.; Fei, Y.; Mitchell, P.; Morozs, N.; Shen, L.; Tozer, T. Data Packet Structure and Modem Design for Dynamic Underwater Acoustic Channels. *IEEE J. Ocean. Eng.* **2019**, *44*, 837–849. [[CrossRef](#)]
- Felemban, E.; Shaikh, F.; Qureshi, U.; Sheikh, A.; Qaisar, S. Underwater Sensor Network Applications: A Comprehensive Survey. *Int. J. Dist. Sens. Netw.* **2015**, *11*, 896832. [[CrossRef](#)]
- Ali, S.; Ashraf, A.; Qaisar, S.B.; Afridi, M.K.; Saeed, H.; Rashid, S.; Felemban, E.A.; Sheikh, A.A. SimpliMote: A Wireless Sensor Network Monitoring Platform for Oil and Gas Pipelines. *IEEE Syst. J.* **2018**, *12*, 778–789. [[CrossRef](#)]
- Morozs, N.; Mitchell, P.D.; Zakharov, Y. Linear TDA-MAC: Unsynchronized Scheduling in Linear Underwater Acoustic Sensor Networks. *IEEE Netw. Lett.* **2019**, *1*, 120–123. [[CrossRef](#)]

9. Lmai, S.; Chitre, M.; Laot, C.; Houcke, S. Throughput-efficient super-TDMA MAC transmission schedules in ad hoc linear underwater acoustic networks. *IEEE J. Ocean. Eng.* **2017**, *42*, 156–174. [[CrossRef](#)]
10. Luque-Nieto, M.; Moreno-Roldán, J.M.; Poncela, J.; Otero, P. Optimal Fair Scheduling in S-TDMA Sensor Networks for Monitoring River Plumes. *J. Sens.* **2016**. [[CrossRef](#)]
11. Diamant, R.; Shirazi, G.N.; Lampe, L. Robust Spatial Reuse Scheduling in Underwater Acoustic Communication Networks. *IEEE J. Ocean. Eng.* **2014**, *39*, 32–46. [[CrossRef](#)]
12. Jiang, S. State-of-the-Art Medium Access Control (MAC) Protocols for Underwater Acoustic Networks: A Survey Based on a MAC Reference Model. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 96–131. [[CrossRef](#)]
13. Heidemann, J.; Stojanovic, M.; Zorzi, M. Underwater sensor networks: Applications, advances and challenges. *Philos. Trans. R. Soc. A* **2012**, *370*, 158–175. [[CrossRef](#)] [[PubMed](#)]
14. Chirdchoo, N.; Soh, W.; Chua, K. MU-Sync: A Time Synchronization Protocol for Underwater Mobile Networks. In Proceedings of the ACM International Workshop on Underwater Networks (WuWNet 2008), San Francisco, CA, USA, 15 September 2008.
15. Molins, M.; Stojanovic, M. Slotted FAMA: A MAC protocol for underwater acoustic networks. In Proceedings of the IEEE OCEANS, Boston, MA, USA, 18–21 September 2006.
16. Noh, Y.; Lee, U.; Han, S.; Wang, P.; Torres, D.; Kim, J.; Gerla, M. DOTS: A Propagation Delay-Aware Opportunistic MAC Protocol for Mobile Underwater Networks. *IEEE Trans. Mob. Comput.* **2014**, *13*, 766–782. [[CrossRef](#)]
17. Diamant, R.; Casari, P.; Campagnaro, F.; Zorzi, M. A Handshake-Based Protocol Exploiting the Near-Far Effect in Underwater Acoustic Networks. *IEEE Wirel. Commun. Lett.* **2016**, *5*, 308–311. [[CrossRef](#)]
18. Syed, A.A.; Ye, W.; Heidemann, J. Comparison and Evaluation of the T-Lohi MAC for Underwater Acoustic Sensor Networks. *IEEE J. Sel. Areas Commun.* **2008**, *26*, 1731–1743. [[CrossRef](#)]
19. Liao, W.H.; Huang, C.C. SF-MAC: A Spatially Fair MAC Protocol for Underwater Acoustic Sensor Networks. *IEEE Sens. J.* **2012**, *12*, 1686–1694. [[CrossRef](#)]
20. Akyildiz, I.F.; Pompili, D.; Melodia, T. Underwater acoustic sensor networks: Research challenges. *Ad Hoc Netw.* **2005**, *3*, 257–279. [[CrossRef](#)]
21. Morozs, N.; Mitchell, P.; Zakharov, Y. TDA-MAC: TDMA without Clock Synchronization in Underwater Acoustic Networks. *IEEE Access* **2018**, *6*, 1091–1108. [[CrossRef](#)]
22. Morozs, N.; Mitchell, P.; Zakharov, Y. Dual-Hop TDA-MAC and Routing for Underwater Acoustic Sensor Networks. *IEEE J. Ocean. Eng.* **2019**, *44*, 865–880. [[CrossRef](#)]
23. Morozs, N.; Gorma, W.; Henson, B.; Shen, L.; Mitchell, P.D.; Zakharov, Y. Channel Modeling for Underwater Acoustic Network Simulation. *IEEE Access* **2020**, *8*, 136151–136175. [[CrossRef](#)]