Proceedings Paper:
Gorma, Wael and Mitchell, Paul Daniel orcid.org/0000-0003-0714-2581 (2019) An Adaptive TDMA-based MAC Protocol for Underwater Acoustic Sensor Networks. In: WUWNET'19: Proceedings of the International Conference on Underwater Networks & SystemsOctober 2019 Article No.: 22 Pages 1–8. ACM , pp. 1-8.

https://doi.org/10.1145/3366486.3366549

An Adaptive TDMA-based MAC Protocol for Underwater Acoustic Sensor Networks

Wael Gorma and Paul D. Mitchell
wmg503@york.ac.uk
paul.mitchell@york.ac.uk
Department of Electronic Engineering, University of York
York, United Kingdom

ABSTRACT

Acoustic propagation through water suffers from attenuation that increases with both signal frequency and transmission range, with time-varying long propagation delays. The power spectral density of underwater ambient noise also changes with frequency. As a result, the usable channel bandwidth is heavily constrained and Medium Access Control (MAC) protocol design for Underwater Acoustic Sensor Networks (UASNs) is challenging. Striking a balance between channel utilisation and network end-to-end delay is particularly difficult. The Combined Free/Demand Assignment Multiple Access (CFDAMA) protocol has been shown to effectively minimise end-to-end delay and maximise channel utilisation, but existing approaches are reliant on synchronisation which is hard to achieve underwater. This paper introduces a novel robust MAC protocol, based on CFDAMA, exclusively designed for UASNs and called CFDAMA-NoClock. It is capable of providing an adaptive MAC solution without the need for synchronisation amongst independent node clocks. The protocol demonstrates a high level of practicality and simplicity. Both analytical models and comprehensive event-driven simulation of several underwater scenarios show that CFDAMA-NoClock can offer excellent delay/utilisation performance under two distinct traffic types and with various network parameters selected based on practical UASN technologies.

CCS CONCEPTS

• **Networks** → Medium Access Protocols;

KEYWORDS

CFDAMA, Medium Access Control, TDMA, Underwater Acoustic Sensor Networks

ACM Reference Format:
Wael Gorma and Paul D. Mitchell. 2019. An Adaptive TDMA-based MAC Protocol for Underwater Acoustic Sensor Networks. In *WUWNet'19: International Conference on Underwater Networks and Systems (WUWNET'19), October 23–25, 2019, Atlanta, GA, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3366486.3366549

WUWNet'19, October 23–25, 2019, Atlanta, GA, USA

ACM ISBN 978-1-4503-7740-9/19/10…$15.00
https://doi.org/10.1145/3366486.3366549

1 INTRODUCTION

Underwater Acoustic Sensor Networks (UASNs) are the means of enabling a wide range of ocean monitoring applications, ranging from scientific and industrial to military and homeland security applications [2]. Underwater acoustic channels are known as a complex and time-variant transmission medium. The propagation of an acoustic signal through water is characterised with attenuation that increases with both the signal frequency and transmission range. The background noise also changes with the frequency. The channel capacity is a function of the transmission range and can be extremely limited [18].

To maximise utilisation of the available capacity, central coordination of a number of acoustic transmissions from different terminals is desirable. Scheduling-based techniques are preferred over other separation techniques. They support adaptive channel capacity allocation and allow variable data rates [5] [13]. However, underwater acoustic channels feature long time-varying propagation delays, leading to temporal and spatial uncertainty [9], the phenomena of space unfairness [3] and momentary connection losses. This brings about scheduling difficulties. The most popular approach is the use of "a global scheduler" offering the requirements of a globally scheduled solution, with the use of guard intervals and the exchange of relative-timing signals. To this end, single-hop topologies are preferred where feasible to minimise the complexity of scheduling algorithms and increase their feasibility. Applications of UASNs are still evolving, and it is envisioned that primarily two types of data traffic will characterise such sensing networks: event-driven and periodic sensing [5] [8]. The two types significantly differ in their traffic patterns. This makes it more challenging to design a single Medium Access Control (MAC) protocol that can be adaptive to various applications.

Given the above facts, UASNs require MAC solutions featuring simplicity and assuring maximum achievable channel utilisation, minimum end-to-end delays, fairness and adaptiveness to traffic changes. The combined round-robin free and demand assignment schemes can provide adaptive TDMA-based MAC solutions, allowing UASNs to alleviate the impact of long propagation delays as well as limited channel capacity, overcome the space unfairness phenomena and adapt to varying traffic demands. For a more robust CFDAMA scheme that can operate reliably despite the potential synchronisation difficulties, this paper introduces CFDAMA without clock synchronisation (CFDAMA-NoClock) scheme. The scheme provides an adaptive TDMA solution enabling back-to-back packet reception at the gateway node without the need of global timing (i.e. a synchronised clock at every node).

The rest of the paper is organised as follows: Section 2 reviews some related work; Section 3 describes the CFDAMA-NoClock scheme; Section 4 presents the simulated underwater scenarios, parameters and illustrates the outcomes of this study; and finally, Section 5 concludes the paper.

2 MOTIVATION AND RELATED WORK

CFDAMA combines two capacity assignment strategies: free assignment and demand assignment. The major advantage of the CFDAMA protocol is that it exploits the contention-less nature of free assignment and the effectiveness of demand assignment in achieving high channel utilisation. This combination can optimise the balance between the end-to-end delay and channel utilisation. Prior to this work, a number of publications introduce several CFDAMA variants as MAC solutions for UASNs. [4] is a study limited to a conventional CFDAMA variant based on a random access request strategy. Following that, [6] introduces a new scheme based on CFDAMA, called CFDAMA with Intermediate Scheduler, which enhances the overall delay/utilisation performance by minimising the average round-trip time between sensor nodes and the CFDAMA central scheduler. Following that, [5] introduces a CFDAMA scheme, namely CFDAMA with Systematic Round Robin requests (CFDAMA-SRR). It increase the effectiveness of CFDAMA underwater by systematising the distribution of request slots based on the location of nodes with respect of the network gateway. This can allow CFDAMA to have a bias against transmissions associated with long round-trip demand assigned slots. When request slots are given to adjacent nodes successively starting from the centre to the edge of the network, positioning of nodes relative to the gateway will have a correlation with the availability of packets in their queues. The farther away the node is, the larger number of packets will be served in a single request opportunity [5].

The functionality of a scheduling-based MAC protocol should be stable despite any synchronisation difficulties. They should be able to operate under long, unknown propagation delays and possible clock drift. [13] introduces an underwater MAC protocol, called Transmit Delay Allocation MAC (TDA-MAC), incorporating a scheduling algorithm that allows a TDMA-like slotted packet reception at the gateway without the need for local synchronisation to a global clock [14]. There, for each data transmission cycle (i.e. TDMA frame), the gateway broadcasts a single packet, REQ packet, to trigger the transmission of one data packet, if any, per sensor node, in a timely manner. In every sensor node, the transmission of a data packet is timed to happen in a certain instructed period of time referenced to the reception of the REQ packet. The developers of TDA-MAC highlight some protocol limitations in utilising the channel; limitations which are overcome via an enhanced variant called Accelerated TDA-MAC (ATDA-MAC) [13]. ATDA-MAC works similarly, but it uses two channels, one of which is dedicated to data packets only, while the other is dedicated for REQ packets.

Like with any other fixed TDMA protocols, achieving high channel utilisation is subject to the presence of a full buffer traffic source. TDA-MAC does not incorporate a mechanism allowing adaptation to the changes in the statistical behaviour of the data traffic source and the instantaneous demand of individual sensor nodes. If every node is enabled to transmit a run of successive packets as demanded after receiving the REQ packet, then the protocol will act as an adaptive TDMA scheme. This paper introduces a new CFDAMA variant inspired by the notion of no clock synchronisation. The primary motivations for this new scheme can be summarised as follows:

- It increases the practicality of CFDAMA by allowing operation with no clock synchronisation amongst nodes.
- It enables high channel utilisation and allows it to approach the theoretical maximum with controlled delay performance.
- It enables instantaneous adaptation to the variation in data traffic conditions in terms of the duration of a specific burst, inter-burst gaps, the duty cycle of bursts and the channel load level.

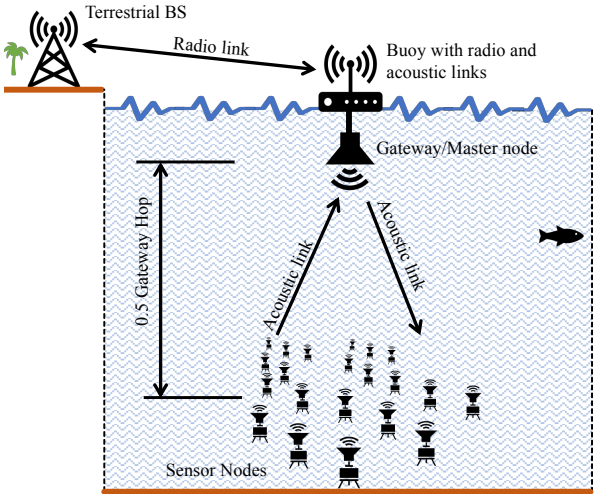


Figure 1: A centralised UASN where the CFDAMA-NoClock protocol is employed

3 THE CFDAMA-NoCLOCK SCHEME

Detailed discussion on CFDAMA can be found in [4][5][6]. With respect to Figure 1, the gateway node, acting as a usual CFDAMA coordinator, needs to be able to estimate the propagation delay to every sensor node and piggyback timing instructions in the acknowledgement packet transmitted during the CFDAMA forward frame [5]. An original CFDAMA variant suitable for this new capability is CFDAMA with Piggy-Backed requests (CFDAMA-PB) [10], which piggybacks capacity requests onto data packets. Being a TDMA-like protocol increases its compatibility with the NoClock scheduling algorithm presented in this paper. The implementation of CFDAMA-PB in the context of ATDA-MAC requires an initial set-up stage prior to the actual data transmissions. In this initial stage, the propagation delays between the gateway and every sensor node are accurately measured via a handshaking technique exchanging PING packets [13]. This process lasts for a short period of time typically of the order of several minutes, depending on the density of nodes and their spatial distribution.

Once the propagation delay is estimated, the scheme can begin the data packet transmission stage. To enable CFDAMA-PB to operate without a synchronised clock, a number of adjustments are

required. The acknowledgement packets sent during the CFDAMA forward frame will be replaced by a packet acting similarly to the REQ packet of TDA-MAC, but with an extra payload. Instead of sending an exclusive packet to every sensor node to inform them with their allocated free or demand slots, a single packet denoted by ACK-REQ is broadcast to inform every node the number of successive slots allocated (N_{rs}) to it and the amount of time, i.e. delay-to-slot (DTS), the node has to wait before it can start a run of successive packet transmissions as allocated. DTS acts similarly to the TDI packet of TDA-MAC, but the Tx delays are calculated differently. The CFDAMA return frame will remain operating as usual, except for the fact that at every sensor node, the transmission of N_{rs} data packet(s) cannot begin until it receives the ACK-REQ and waits the amount of time that is stated in the DTS segment of the ACK-REQ packet. This process is depicted in Figure 2. There, the gateway broadcasts an ACK-REQ packet on the forward channel to be received by every node at different arrival times due to them being positioned at different locations. Upon the arrival of the ACK-REQ packet, the concerned node waits the appropriate amount of time, and then, transmits a run of N_{rs} data packets. On condition that the appropriate CFDAMA frame length and forward frame delay are used, this process leads to CFDAMA-like packet arrival at the gateway without the need for clock synchronisation amongst sensor nodes. This packet reception timing is illustrated in Figure 2.

3.1 CFDAMA-NoClock: Calculating Delay-to-Slots

The gateway node constructs the DTS segment of the ACK-REQ packet that needs to be transmitted to every sensor node on a frame-by-frame basis in order to assign the Tx delays. The Tx delay for the n^{th} node, where $n = 2, 3, \ldots, N$ is given by:

$$\tau_{\text{tx}}[n] = 2(\tau_{\text{p}}[N] - \tau_{\text{p}}[n]) + \sum_{i=1}^{n-1} (N_{\text{as}}[i]\tau_{\text{slot}}) \tag{1}$$

where $\tau_{\text{p}}[n]$ is the propagation delay from the gateway to the n^{th} sensor node, N^{th} sensor node is the farthest node from the gateway, $\tau_{\text{tx}}[n]$ is the delay-to-slot assigned to the n^{th} sensor node, N_{as} is the number of data slots assigned to the n^{th} node in the current frame and τ_{slot} the duration of data slot which must satisfy the following constraint:

$$\tau_{\text{slot}} \geqslant T_{\text{dp}}[n] + T_{\text{g}}[n] \tag{2}$$

Where $T_{\text{dp}}[n]$ is the duration of the n^{th} node's data packet including the segment of its capacity requests and $T_{\text{g}}[n]$ is the guard interval after the n^{th} node's data packet reception at the gateway. In every round of transmitting an ACK-REQ packet, two vectors: $\boldsymbol{\tau}_{\text{tx}} = (\tau_{\text{tx}}[1], \tau_{\text{tx}}[2], \ldots, \tau_{\text{tx}}[N])$ and $\boldsymbol{N}_{\text{as}} = (N_{\text{as}}[1], N_{\text{as}}[2], \ldots, N_{\text{as}}[N])$ are constructed at the gateway node, sorted based on the n^{th} node's location from nearest to farthest from the gateway and loaded onto the ACK-REQ packet. In the absence of slot requests, the scheduler assigns a free slot to every sensor node, for example, the case of the first round in which capacity requests have not been made at this point in time. The gateway is able to periodically assess the accuracy of the measured propagation delays by comparing the expected and the actual time of arrival of data packets. If error values exceed a certain sustainable limit, the gateway node can then update the the Tx delays accordingly.

Algorithm 1 CFDAMA-NoClock algorithm implementation on the gateway node; ACK-REQ - CFDAMA acknowledgement and data request packet

1: **for** every sensor node ($n = 1, 2, 3, \ldots N$) **do**
2: Transmit PING packet to n^{th} sensor node
3: Wait for PING packet back from n^{th} sensor node
4: Calculate propagation delay $\tau_{\text{p}}[n]$ to n^{th} sensor node
5: **end for**
6: Calculate Tx delay $\tau_{\text{tx}}[n]$ for every n using (1)
7: Determine the demand/free slot allocation according to CFDAMA-PB rules
8: Construct τ_{tx} and N_{as} vectors and load them onto ACK-REQ packet
9: Broadcast the ACK-REQ packet
10: **while** CFDAMA slot jitter is below threshold (no collisions) **do**
11: Measure the errors between expected and actual data packet arrivals
12: **if** CFDAMA slot jitter is above a threshold **then**
13: Compensate for propagation delay estimation errors using the actual value
14: Go to Step 6
15: **end if**
16: **end while**

Algorithm 2 CFDAMA-NoClock algorithm implementation on a sensor node; TDI - ACK-REQ - CFDAMA acknowledgement and data request packet

1: **if** PING packet received from gateway node **then**
2: Transmit PING packet back to gateway node
3: **end if**
4: **if** ACK-REQ packet received from gateway node **then**
5: Schedule packet transmission with allocated delay and for N_{as} successive data packets
6: **end if**

Algorithm 1 shows the implementation steps taken at the gateway node to run the CFDAMA-NoClock protocol. Algorithm 2 shows the implementation steps taken at every sensor node to operate in accordance with the proposed protocol. The complexity and computational requirements are low at the sensor nodes; the algorithm demonstrates these vital features with only two basic reactive operations. Most of the processing requirements of the scheme are at the gateway node.

3.2 CFDAMA-NoClock: Scheduling the CFDAMA Forward Frame

Following the process of measuring all propagation delays to the sensor nodes, the gateway has to establish the offset time between the CFDAMA forward and return frames. In other words, it has to determine during the current return frame the proportion of slots

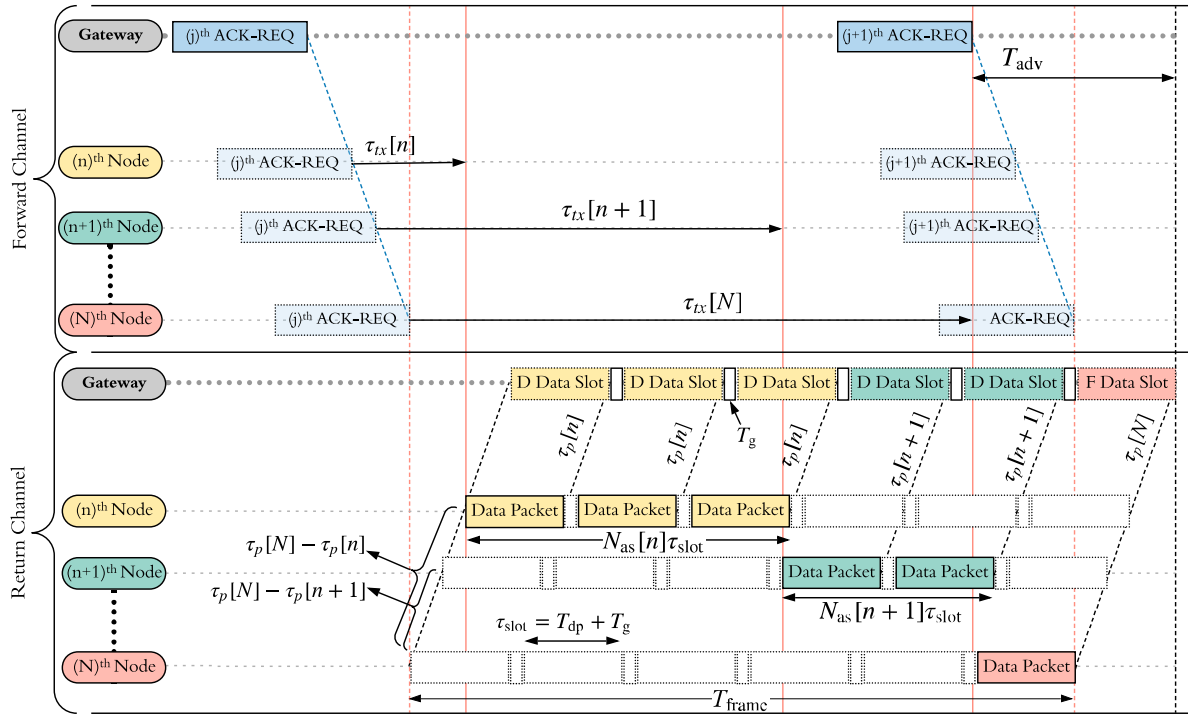


Figure 2: An arbitrary CFDAMA-NoClock transmission cycle with its two channels working concurrently; ACK-REQ - acknowledgement and data request packet

over which the transmission of the next ACK-REQ packet takes precedence. For instance, the illustrative chronology in Figure 2 shows that the gateway brings forward its ACK-REQ packet transmission by more than 1.5 data slots. This proportion is denoted by T_{adv} in the following description. Theoretically, the larger the T_{adv} is, the smaller the gap between successive CFDAMA return frames will be, which results in better channel throughput. However, a constraint has to be satisfied in order not to over accelerate the next return frame and cause frame overlap with the current return frame at the gateway:

$$T_{adv} = \max_{n=1\ldots N} \left\{ n \,|n\tau_{slot} \leqslant 0.5\tau_p[N] \right\} \tag{3}$$

where $T_{g,rp}$ is the guard interval between the transmission of an ACK-REQ packet and the adjacent data packet reception. This constraint ensures that, for every slot over which the transmission of ACK-REQ takes precedence, the new ACK-REQ packet does not arrive at the concerned node, which will utilise this slot, before it completes the transmission of the previous data packet.

3.3 CFDAMA-NoClock: Optimal CFDAMA Frame Length

Whilst the maximum limit of the CFDAMA return frame interval is extendible based on the desired delay/utilisation performance, its shortest interval has a certain limit given the no-synchronised-clock circumstances. Ideally, the gateway node is required to transmit at least one broadcast ACK-REQ packet during the interval over which the data packets from all sensor nodes assigned capacity are received. This interval will then be the minimum duration of the CFDAMA return frame T_{frame}. Therefore, this T_{frame} has to satisfy the two constraints:

$$\begin{aligned} T_{frame} &\geq T_{min,delay} \\ T_{frame} &\geq T_{min,demand} \end{aligned} \tag{4}$$

where $T_{min,delay}$ is the constraint placed by the longest round-trip propagation delay, $\tau_p[N]$, between the gateway and the sensor nodes, and $T_{min,demand}$ is another important constraint placed by the channel in terms of its data carrying capacity, i.e. if the former constraint is not the limiting factor, the latter is, in which case the performance is limited by the packet duration, capacity demand and statistical behaviour of the data traffic source.

The first constraint $T_{min,delay}$ is calculated using the following expression:

$$T_{frame} \geq \max_n \tau_p[n] - \min_n \tau_p[n] \tag{5}$$

where T_{frame} is the frame interval. This expression states that the frame length cannot be shorter than the difference between the longest and the shortest round-trip delays to ensure receiving at least one data packet, if any, from the farthest sensor node.

In the second case, the data carrying capacity constraint on T_{frame} is given by the following expression:

$$T_{min,demand} = \sum_{n=1}^{N} N_{as}[n] \left(T_{dp}[n] + T_g[n] \right) \tag{6}$$

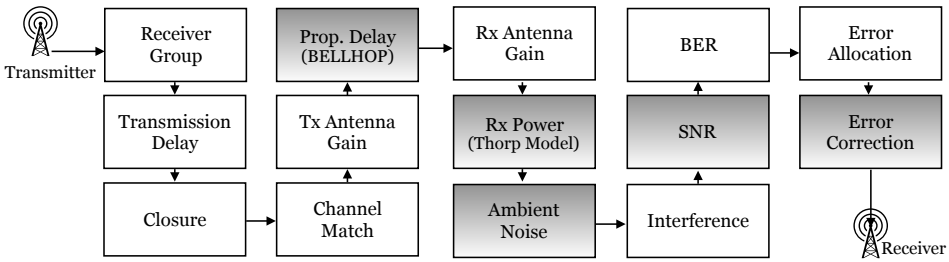


Figure 3: Riverbed-based underwater acoustic channel [5]

The expression ensures that the minimum T_{frame} should not be smaller than the duration of all data packets of the current cycle plus the guard intervals amongst them. Taking both constraints into account, the minimum possible interval between two consecutive ACK-REQ packet transmissions can then be expressed as:

$$T_{\text{frame,min}} = \max\left(T_{\text{min,delay}}, T_{\text{min,demand}}\right) \tag{7}$$

In practice, this frame interval is typically specified to a given application based on how frequently the sensor readings require gathering. The CFDAMA-PB scheme can give its best delay/utilisation performance when $N\tau_{\text{slot}}$ is set to be close to the $2\max_n \tau_{\text{p}}[n]$, composed of N_{slots} data slots where $N_{slots} \geqslant N$. This ensures that a larger number of nodes, if not all nodes, can make a capacity request in every return frame. The suitable frame and data-slot durations are chosen on the basis of the desired channel capacity and transmission rate of a given application taking into consideration constraint 7.

3.4 CFDAMA-NoClock: Achievable Channel Utilisation

The maximum achievable channel utilisation of CFDAMA-NoClock is limited to the maximum channel utilisation of the no-synchronised-clock algorithm and CFDAMA-PB capacity overhead, which can be approximated as follows:

$$\gamma_{max} = \frac{(1-\vartheta)\sum_{n=1}^{N_{\text{slots}}} T_{\text{dp}}[n]}{\sum_{n=1}^{N_{\text{slots}}} \left(T_{\text{dp}}[n] + T_{\text{g}}[n]\right)} \tag{8}$$

Where ϑ is the fraction of packet overhead due to the embedded capacity requests. This expression states that the guard intervals and packet overheads are the primary cause of throughput loss. Guard intervals are an essential design parameter allowing adaptive timing to accommodate typical motion of "fixed" nodes. For example, a 100 ms guard period between data packets can allow tolerance to the changes in a node location of up to 150 m before propagation delays require re-estimating. Achieving this throughput is conditional on the elimination of any potential gaps interleaving adjacent CFDAMA-PB frames (i.e. gaps separating every set of packet reception). To achieve this optimal throughput performance, the interval of the frame should be at least $2\tau_{\text{p}}[N]$, and the proportion of data slots over which the transmission of the next ACK-REQ packet takes precedence should be: $T_{\text{adv}} = 0.5\tau_{\text{p}}[N]$.

4 PERFORMANCE EVALUATION OF CFDAMA-NoCLOCK

In this section, the performance of CFDAMA-NoClock is evaluated using a CFDAMA simulation model developed in Riverbed Modeller [7] for the UASN model depicted in Figure 1. The performance of the proposed scheme is compared with the performance of both standard CFDAMA-PB and ideal synchronised TDMA, under both periodic data gathering obeying Pareto ON/OFF traffic and random Poisson traffic conditions. The simulation setup is described in this section.

4.1 Simulation Setup

4.1.1 Underwater acoustic channel model. The simulated acoustic link works based on the Riverbed Modeller stages shown in Figure 3. The BELLHOP program [16] is employed to provide the acoustic links with the actual acoustic propagation delay based on a realistic Sound Speed Profile (SSP) of a case derived by Dushaw [1] from the 2009 World Ocean Atlas temperature, pressure and salinity data at $(56.5^{o}N, 11.5^{o}W)$ in April, i.e. around the North Atlantic Ocean off the coast of the UK and Ireland. An empirical model [17] is used to predict the underwater ambient noise. The Thorp model [19] is used to determine the absorption coefficient, used to estimate the received power. The fourteen Riverbed Modeller stages are executed on a per-receiver basis whenever a packet is transmitted. Through these stages, the signal to noise ratio (SNR) of each received packet is calculated, and subsequently, the Bit Error Rate (BER) is estimated using a look-up table. Both the proportion of bit errors due to noise and the level of interference with other packets determine the packet's eligibility for successful reception at its receiver. The receiver rejects all packets involved in overlapped arrivals and the packets whose number of bit errors exceeds a certain threshold.

4.1.2 Network Topology and Simulation Parameters. With reference to Figure 1, several scenarios of various network sizes (20, 50 and 100 nodes) and packet durations are investigated. In the simulator, sensor nodes are deployed randomly over an area of $6 \times 6\ km$ with a centralised gateway at a 20m depth. The depths of sensor nodes are uniformly distributed between 470 and 490 m. These parameters corresponds to a typical oil reservoir seismic monitoring scenario, e.g. [12]. They are also within the range of operating parameters of some acoustic modems, e.g. the EvoLogics S2CR 15/27 modem [15]. These scenarios allow different test options for performance evaluation and enable comparison with other

Table 1: Simulation parameters

Attribute	Value
Transmission Range	6×6 km
Number of Nodes	20, 50 or 100
Bandwidth	30 kHz
Data Rate	9600bps
Packet Size	64, 256, 512 bits
Packet Duration	6.66, 26.66, 53.33 ms
Request Slot Size	8 bit
Request Slot Duration	0.833 ms
Number of Data Slots in Frame	650, 256 and 128
Traffic Load Range	0.1 - 1 Erlangs

approaches in the literature. The simulation parameters are listed in Table 1. In all the results presented, channel load refers to the level of demand placed on a channel, measured in Erlangs and expressed as a percentage of the overall data carrying capacity of the channel. To reflect on the wide-ranging underwater applications, two distinct traffic models (Poisson OFF and Pareto ON/OFF) [11] are developed in Riverbed Modeller for this performance evaluation.

4.2 Analysis of The Results

Figure 4 is a bar chart representing the network throughput achieved at the gateway working as a sink receiving collected data from all sensor nodes. Here, throughput is defined as the proportion of the successful data transmission that is effectively used to transfer new information after an amount of traffic is placed on the channel, expressed as a percentage of the channel capacity, i.e the overall data carrying capacity. The chart shows the throughput performance of CFDAMA-NoClock and the synchronised CFDAMA-PB for a baseline comparison. It also compares with the analytical predictions of the optimal CFDAMA-NoClock throughput given by Equation (8).

The results in Figure 4 indicate that there is a negligible difference in throughput performance between CFDAMA-NoClock and synchronised CFDAMA-PB in all three simulated scenarios. This demonstrates that CFDAMA-NoClock can achieve the performance of ideal CFDAMA-PB without the need for a synchronised clock in every sensor node. The primary source of capacity waste in the case of CFDAMA-NoClock is the guard intervals amongst data packets, suggested as 5% of the data packet length. In practice, the length of the guard interval can be set to be a more realistic value suited to a given network deployment experiencing certain motion of nodes, and/or propagation delay jitter.

Furthermore in Figure 4, the comparison with the analytically predicted values of the network throughput indicates that Equation (8) provides a good performance estimate based on the system parameters, e.g. packet duration, guard intervals and packet overhead. The very slight disagreement between the analytically predicted optimal throughput performance of CFDAMA-NoClock and the simulation outcome is attributable to the collection time, i.e. the time between the very first ACK-REQ packet and the subsequent set of data collection (i.e. first CFDAMA return frame). This inevitable gap in channel utilisation cannot be filled due to not being preceded by any data packets and followed by the first set

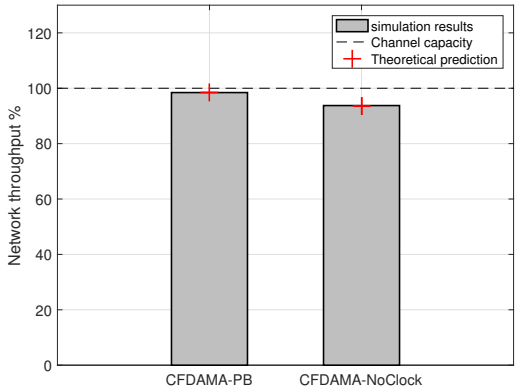
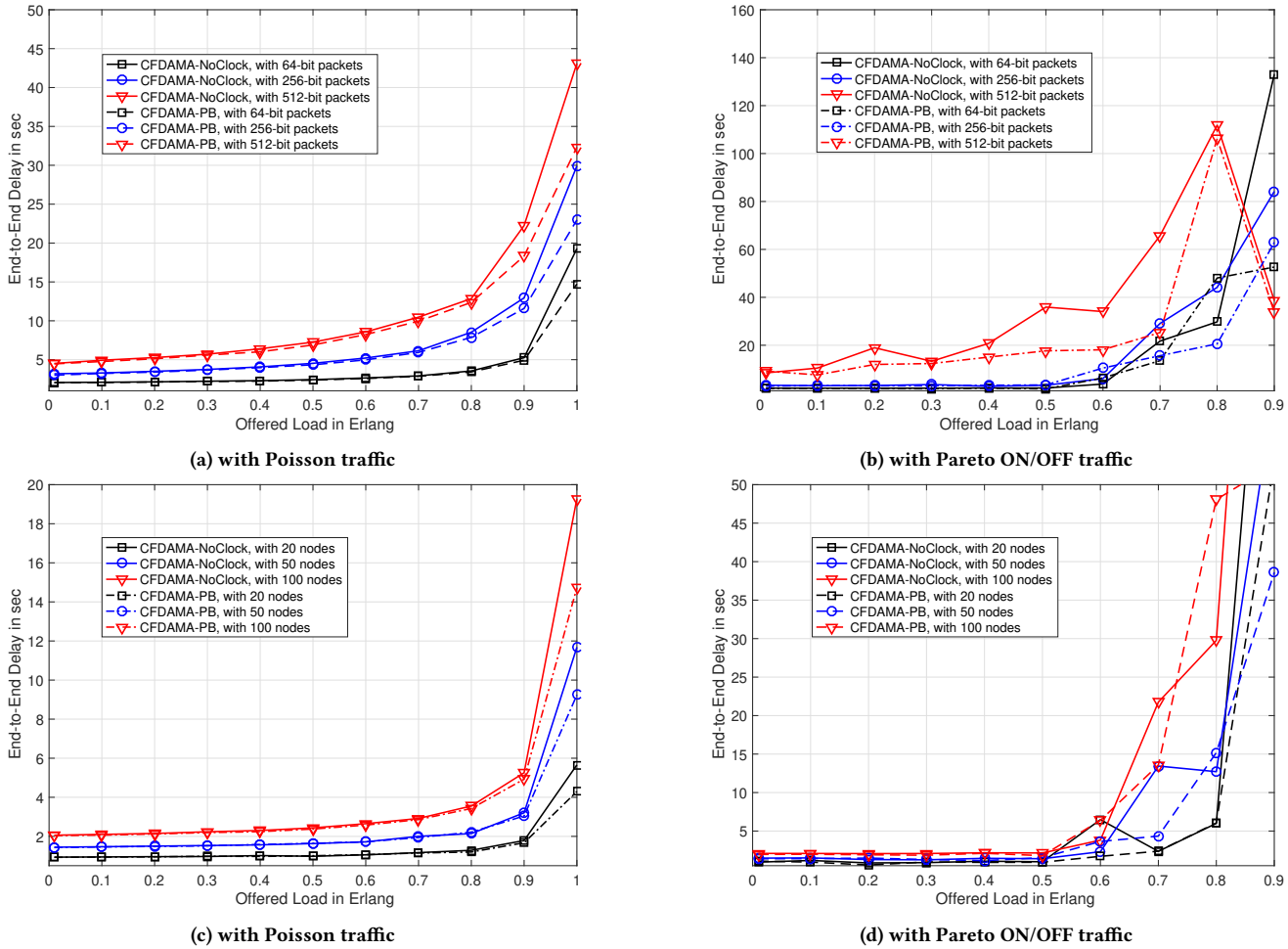


Figure 4: Network throughput achieved by CFDAMA-NoClock and CFDAMA-PB under Poisson data traffic. The simulation results are compared with the analytical prediction given by Equation (8)**; packet size 512 bit and data rate 9.2 kbit/s**

of data packet reception. This gap is proportional to the longest round-trip propagation delay.

Another important performance metric is the end-to-end delay achieved by our NoClock scheme under the two distinct traffic conditions, i.e. Poisson and Pareto ON/OFF. CFDAMA-NoClock should not experience very different delay/utilisation performance from the typical CFDAMA-PB delay performance. Figure 5 shows the delay/utilisation performance of CFDAMA-NoClock with the two traffic types (Poisson and Pareto ON/OFF), different numbers of nodes (20, 50 and 100) and different data packet sizes (64, 256 and 512 bits) at a data rate 9.2 kbit/s. In general, the mean end-to-end delay starts to increase exponentially as the offered load value approaches the maximum network throughput of a given scenario. Like CFDAMA-PB, the CFDAMA-NoClock scheme is still capable of providing the expected end-to-end delay performance with both traffic types. At low to medium channel loads, the mean end-to-end delay with both traffic models is similar. When the channel load is 50% of the channel capacity, the end-to-end delay is much higher with the Pareto ON-OFF traffic. With full channel load, 100 nodes and different packet sizes (64, 256 and 512 bit), the shortest mean end-to-end delay achieved with the shortest packet duration of 6.66 ms. There, as Figure 5a-5b shows, a data packet can be collected approximately every 19 s with Poisson traffic and 135 s with Pareto ON/OFF. With full channel load and different numbers of nodes, the shortest mean end-to-end delay is achieved with the smallest network size of 20 nodes. There, as Figure 5c-5d shows a data packet can be collected approximately every 5.8 s on average with Poisson traffic and 96 s with Pareto ON/OFF.

CFDAMA-NoClock vs. TDMA. With a moderate traffic load, the classical TDMA MAC protocol is a good solution in terms of throughput as data packets are transmitted without MAC overhead. However, its delay/utilisation performance is dependent on the accuracy of node clock synchronisation. Figure 6 shows the mean end-to-end



(a) with Poisson traffic

(b) with Pareto ON/OFF traffic

(c) with Poisson traffic

(d) with Pareto ON/OFF traffic

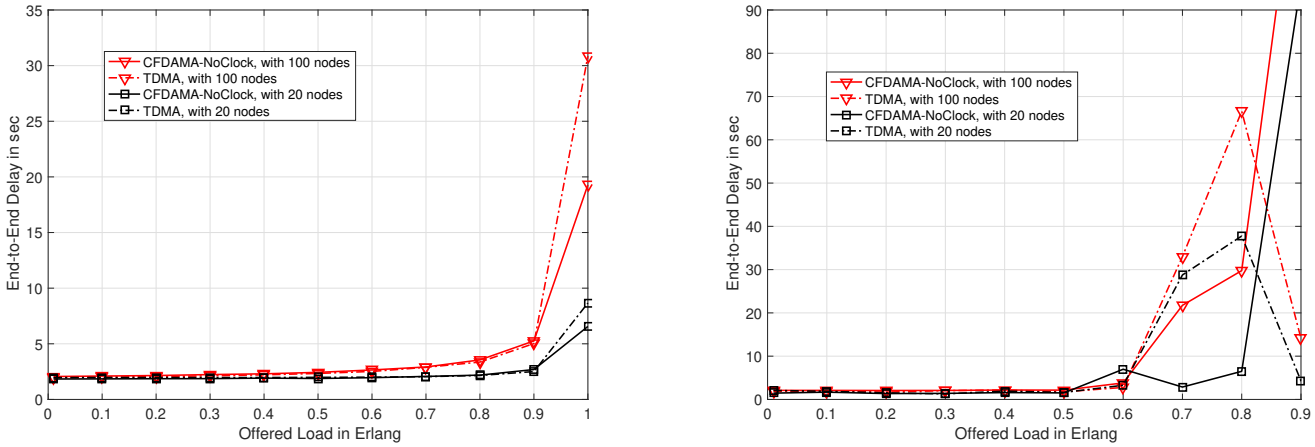
Figure 5: The delay/utilisation performance of CFDAMA-NoClock vs CFDAMA-PB with different 20, 50 and 100 nodes; 64, 256 and 512 bit packets and 9.2 kbit/s; and two distinct traffic types

delay performance of the CFDAMA-NoClock scheme against the performance of the ideal synchronised TDMA protocol with both Poisson and Pareto ON/OFF traffic types offered by 20 and 100 nodes. The results in Figure 6a indicate that with moderate channel loads, both schemes perform almost the same with 20 and 100 nodes. This is attributable to the limited burstiness of the Poisson traffic which is unable to offer substantial demands for an excessive period of time long enough to allow the demand assigned slots of CFDAMA to contribute effectively. In this instance, the free assignment scheme, underlying CFDAMA, contributes more effectively and the performance is similar to TDMA. At high offered load values, CFDAMA-NoClock has a small advantage over TDMA in terms of end-to-end delay. This is attributable to the increased demand made for packets and the fact that the TDMA slots, assigned periodically, cannot be as effective as the on-demand slots assigned by CFDAMA-NoClock at such high load levels. At a high channel load of 90% and 100 nodes, the mean end-to-end delays are around 13 s with CFDAMA-NoClock, and above 20 s with TDMA.

With the Pareto ON/OFF traffic source, increasing the traffic load level causes a significant increase in the spread of end-to-end delay values between the two schemes with a considerable increase in the proportion of packets experiencing very long end-to-end delay. TDMA begins to suffer from instability after the channel load exceeds 80%. At such very high channel load, the traffic sources become able to offer instantaneous load levels exceeding the channel capacity over a significant periods of time. During these periods, packets continue to build up in the sensor node queues until the load level drops below the channel capacity. The CFDAMA-NoClock scheme can cope with such statistical variations in the traffic source level up to higher channel load levels.

5 CONCLUSIONS

This paper introduced a new robust MAC solution for UASNs. It is referred to as CFDAMA-NoClock (CFDAMA without clock synchronisation). It is based on CFDAMA and is a more feasible MAC solution when synchronisation amongst node clocks cannot be attained.



(a) with Poisson traffic **(b) with Pareto ON/OFF traffic**

Figure 6: Comparative delay/utilisation performance of CFDAMA-NoClock vs. TDMA with 64-bit data slots

The elimination of synchronised clocks increases the practicality of this scheme. The MAC operations required to be processed on the sensor nodes are minimal. All the complexity associated with the functionality of the scheme is at the gateway node. The CFDAMA-NoClock scheme exhibits excellent delay/utilisation performance superiority over the performance of ideal synchronised TDMA. Comprehensive event-driven Riverbed simulations of a network deployed on the sea bed show that the proposed protocol is able to closely match its underlying scheme CFDAMA-PB with the advantage of independent unsynchronised node clocks. In all simulated scenarios, ranging from a network of 20, 50 to 100 nodes and packet sizes from 64, 256 to 512 bits, with a data rate of 9.2 kbit/s and two data traffic types (Poisson and Pareto ON/OFF), the proposed protocol achieves the expected very close delay/utilisation performance to that can be achieved with the underlying CFDAMA variant.

REFERENCES

[1] Dushaw B. 2009. Worldwide sound speed, temperature, salinity, and buoyancy from the NOAA world ocean atlas. *Available: http://staff.washington.edu/dushaw/WOA/* (2009).

[2] Emad Felemban, Faisal Karim Shaikh, Umair Mujtaba Qureshi, Adil A. Sheikh, and Saad Bin Qaisar. 2015. Underwater Sensor Network Applications: A Comprehensive Survey. *International Journal of Distributed Sensor Networks* 11, 11 (2015), 896832. https://doi.org/10.1155/2015/896832 arXiv:https://doi.org/10.1155/2015/896832

[3] Chane L. Fullmer and J. J. Garcia-Luna-Aceves. 1995. Floor Acquisition Multiple Access (FAMA) for Packet-radio Networks. *SIGCOMM Comput. Commun. Rev.* 25, 4, 262–273. https://doi.org/10.1145/217391.217458

[4] Wael Gorma and Paul D Mitchell. 2017. Performance of the Combined Free/Demand Assignment Multiple Access Protocol via Underwater Networks. In *Proceedings of the International Conference on Underwater Networks & Systems (WUWNET'17)*. ACM, New York, NY, USA, Article 5, 2 pages. https://doi.org/10.1145/3148675.3148678

[5] Wael Gorma, Paul D Mitchell, Nils Morozs, and Yuriy V Zakharov. 2019. CFDAMA-SRR: A MAC Protocol for Underwater Acoustic Sensor Networks. *IEEE Access* 7 (2019), 60721–60735. https://doi.org/10.1109/ACCESS.2019.2915929

[6] Wael Gorma, Paul D Mitchell, and Yuriy V Zakharov. 2019. CFDAMA-IS: MAC Protocol for Underwater Acoustic Sensor Networks. In *Broadband Communications, Networks, and Systems*, Victor Sucasas, Georgios Mantas, and Saud Althunibat (Eds.). Springer International Publishing, Cham, 191–200.

[7] IS Hammoodi, BG Stewart, A Kocian, and SG McMeekin. 2009. A Comprehensive Performance Study of OPNET Modeler for ZigBee Wireless Sensor Networks. In *2009 Third International Conference on Next Generation Mobile Applications, Services and Technologies*. 357–362. https://doi.org/10.1109/NGMAST.2009.12

[8] Albert F. Harris, III, Milica Stojanovic, and Michele Zorzi. 2006. When Underwater Acoustic Nodes Should Sleep with One Eye Open: Idle-time Power Management in Underwater Sensor Networks. In *Proceedings of the 1st ACM International Workshop on Underwater Networks (WUWNet '06)*. ACM, New York, NY, USA, 105–108. https://doi.org/10.1145/1161039.1161061

[9] C-C Hsu, K-F Lai, C-F Chou, and KC-J Lin. 2009. ST-MAC: Spatial-Temporal MAC Scheduling for Underwater Sensor Networks. In *IEEE INFOCOM 2009*. 1827–1835. https://doi.org/10.1109/INFCOM.2009.5062103

[10] Tho Le-Ngoc and I Mohammed Jahangir. 1998. Performance analysis of CFDAMA-PB protocol for packet satellite communications. *IEEE Transactions on Communications* 46, 9 (Sep. 1998), 1206–1214. https://doi.org/10.1109/26.718562

[11] Tozer Tim C Mitchell, Paul D and David Grace. 2000. Improved medium access control for data traffic via satellite using the CFDAMA protocol. In *IEE Seminar on Broadband Satellite: The Critical Success Factors - Technology, Services and Markets*. 18/1–18/7. https://doi.org/10.1049/ic:20000542

[12] Arupa K Mohapatra, Natarajan Gautam, and Richard L Gibson. 2013. Combined Routing and Node Replacement in Energy-Efficient Underwater Sensor Networks for Seismic Monitoring. *IEEE Journal of Oceanic Engineering* 38, 1 (Jan 2013), 80–90. https://doi.org/10.1109/JOE.2012.2208850

[13] Nils Morozs, Paul D Mitchell, and Yuriy V Zakharov. 2018. TDA-MAC: TDMA Without Clock Synchronization in Underwater Acoustic Networks. *IEEE Access* 6 (2018), 1091–1108. https://doi.org/10.1109/ACCESS.2017.2777899

[14] Vincent Ngo, Isaac Woungang, and Alagan Anpalagan. 2014. A schedule-based medium access control protocol for mobile wireless sensor networks. *Wireless Communications and Mobile Computing* 14, 6 (2014), 629–643. https://doi.org/10.1002/wcm.2220

[15] Chiara Petrioli, Roberto Petroccia, Jon Shusta, and Lee Freitag. 2011. From underwater simulation to at-sea testing using the ns-2 network simulator. In *OCEANS 2011 IEEE - Spain*. 1–9. https://doi.org/10.1109/Oceans-Spain.2011.6003638

[16] Michael B Porter. 2011. The BELLHOP manual and user's guide. *Preliminary draft. See http://oalib. hlsresearch. com/Rays/HLS-2010–1. pdf (accessed 1 Sep 2019)* (2011).

[17] Milica Stojanovic. 2007. On the Relationship Between Capacity and Distance in an Underwater Acoustic Communication Channel. *SIGMOBILE Mob. Comput. Commun. Rev.* 11, 4 (Oct. 2007), 34–43. https://doi.org/10.1145/1347364.1347373

[18] Milica Stojanovic and James Preisig. 2009. Underwater acoustic communication channels: Propagation models and statistical characterization. *IEEE Communications Magazine* 47, 1 (January 2009), 84–89. https://doi.org/10.1109/MCOM.2009.4752682

[19] William H Thorp. 1965. Deep-Ocean Sound Attenuation in the Sub-and Low-Kilocycle-per-Second Region. *The Journal of the Acoustical Society of America* 38, 4 (1965), 648–654. https://doi.org/10.1121/1.1909768