12-18-2015

Real System Features and Implications in Underwater Acoustic Networks

Lina Pu
University of Connecticut - Storrs, lina.pu@engr.uconn.edu

Real System Features and Implications in Underwater Acoustic Networks

Lina Pu, Ph.D.

University of Connecticut, 2015

As an emerging research area, underwater acoustic network (UAN) has attracted tremendous interests from both academia and industry in recent years. However, little work has been conducted to test algorithms and protocols in real sea experiments. Due to the complexity of acoustic environments and the uncertainties in acoustic systems, it is difficult for theoretical studies or simulations to evaluate UANs in the real world. Studying real system features of UANs has become crucial in the field. In this dissertation, I study the real system features revealed in sea tests and analyze their impact on underwater media access control (MAC), time synchronization and secret key generation protocols. It covers four research thrusts.

First, through sea experiments, I identify several unique features of real systems. My findings include the long preamble of acoustic modems, heterogeneous packet delivery, communication range uncertainty, multi-hop interference, and delayed data transmission. I analyze these new findings, in hope of shedding some light on the practical design of real underwater networks.

Second, I analyze and evaluate representative MAC protocols in sea tests. Based on the field test results, I study the advantages, shortcomings and limitations of different MAC mechanisms and how they work in real systems. I also propose a practical MAC

design for UANs.

Third, I evaluate representative time synchronization schemes in a lab environment. Based on the experiment results, I analyze the temporal and statistical features of message delivery delay and provide some guidelines on practical time synchronization protocol design and performance improvement.

Fourth, I evaluate the performance of representative RSS based key generation approaches for underwater secure communications. Via sea experiments, I test and analyze how underwater system features affects RSS based key generation protocols. Meanwhile, solutions to improve the performance in terms of key generation rate, randomness and key agreement probability are provided.

Real System Features and Implications in Underwater Acoustic Networks

Lina Pu

B.E., Northwestern Polytechnical University, 2009

M.S., University of Connecticut, 2015

A Dissertation

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

at the

University of Connecticut

2015

APPROVAL PAGE

Doctor of Philosophy Dissertation

Real System Features and Implications in Underwater Acoustic Networks

Presented by

Lina Pu, B.E., M.S.

Major Advisor _____
Jun-Hong Cui

Associate Advisor _____
Swapna Gokhale

Associate Advisor _____
Chun-Hsi Huang

Associate Advisor _____
Zheng Peng

University of Connecticut

2015

ACKNOWLEDGEMENTS

This dissertation was a long time in the making and it would not have been completed, if not for all the support, encouragement, and assistance from my advisors, lab mates, friends, and family.

First of all, I would like to express my heartfelt gratitude to my major advisor, Dr. Jun-Hong Cui, for her excellent guidance, continuous support, generous encouragement and opportunities given to me to practice and grow up throughout my Ph.D. study. All the works could not be accomplished without her persistent and patient supervision. What she taught have benefitted me a lot and will continue benefitting me for my whole life.

I would like to thank my co-advisors Dr. Swapna Gokhale, Dr. Chun-Hsi Huang, and Dr. Zheng Peng for their advice and broad knowledge. It was my great pleasure working with these intelligent and responsible professors. Also I would like to thank Dr. Zaihan Jiang, Naval Research Lab, Washington, DC, for his help on the system feature study on UANs.

I would like to extend my gratitude to my lab mates, Yu Luo, Zhong Zhou, Liu Jun, Yibo Zhu, Haining Mo, Son Le, Hao Zhou, Lei Wan, Li Wei, Robert Martin, Zigeng Wang, Xiaoyan Lu, Fei Dou and Xia Xiao, for their discussions and comments on my research. My thanks for friendship also go to these lab mates and all my friends, for their endless support whenever I needed.

Last, but not least, I give my deepest gratitude and love to my parents and my husband, who have been a constant source of happiness and motivation for me. Their love and support have been my inspiration to accomplish the doctoral program. To my family, I dedicate this dissertation.

PUBLICATIONS

Lina Pu, Yu Luo, Haining Mo, Son Le, Zheng Peng, Jun-Hong Cui, and Zaihan Jiang, "Comparing underwater MAC protocols in real sea experiments," *Computer Communications*, vol. 56, pp. 47–59, 2015.

Lina Pu, Yu Luo, Haining Mo, Zheng Peng, Jun-Hong Cui, and Zaihan Jiang, "Comparing underwater MAC protocols in real sea experiment," In *Proceedings of the IFIP Networking Conference*, 2013, pp. 1–9. IEEE, 2013. (**Best Paper Award**)

Lina Pu, Yu Luo, Zheng Peng, Haining Mo, and Jun-Hong Cui, "Traffic estimation based receiver initiated MAC for underwater acoustic networks," In *Proceedings of the ACM International Conference on Underwater Networks and Systems (WUWNet)*, pp. 1–6, ACM, 2013.

Lina Pu, Yu Luo, Yibo Zhu, Zheng Peng, Jun-Hong Cui, Shruti Khare, Lei Wang, and Benyuan Liu, "Impact of real modem characteristics on practical underwater MAC design," In *Proceedings of the OCEANS*, 2012, pp. 1–6, IEEE, 2012.

Yu Luo, **Lina Pu**, Zheng Peng, and Jun-Hong Cui, "Dynamic control channel MAC for underwater cognitive acoustic networks," In *Proceedings of the International Conference on Computer Communications (INFOCOM)*, IEEE, 2016 (Accepted).

Yu Luo, **Lina Pu**, Zheng Peng, and Zhijie Shi, "RSS-based secret key generation in underwater acoustic networks: advantages, challenges and performance improvements," *IEEE Communications Magazine*, 2016 (Accepted).

Yu Luo, **Lina Pu**, Zheng Peng, Zhong Zhou, and Jun-Hong Cui, "An efficient MAC protocol for underwater multi-user uplink communication networks," *Ad Hoc Networks*, 2015, DOI: 10.1016/j.adhoc.2015.01.011.

Yu Luo, **Lina Pu**, Michael Zuba, Zheng Peng, and Jun-Hong Cui, "Cognitive acoustics: making underwater communications environment-friendly," In *Proceedings of the International Conference on Underwater Networks & Systems (WUWNet)*, pp. 48. ACM, 2014

Yu Luo, **Lina Pu**, Michael Zuba, Zheng Peng, and Jun-Hong Cui, "Challenges and opportunities of underwater cognitive acoustic networks," *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 2, pp. 198211, 2014.

Yu Luo, **Lina Pu**, Zheng Peng, Yibo Zhu, and Jun-Hong. Cui, "ISM: an efficient spectrum management system for underwater cognitive acoustic networks," In *Proceedings of International Conference on Sensing, Communication, and Networking (SECON)*, pp. 414–422, IEEE, 2014.

Yibo Zhu, **Lina Pu**, Zigeng Wang, Xiaoyan Lu, Robert Martin, Yu Luo, Zheng Peng, Jun-Hong Cui, "Underwater acoustic network protocol stacks: simulator-based vs. OS-based", In *Proceedings Of IEEE OCEANS*, 2014

Yu Luo, **Lina Pu**, Zheng Peng, Zhong Zhou, and Jun-Hong Cui, "Effective relay selection for underwater cooperative acoustic networks," In *Proceedings of the International Conference on Mobile Ad-Hoc and Sensor Systems (MASS)*, pp. 104–112. IEEE, 2013.

Haining Mo, **Lina Pu**, Yibo Zhu, Zheng Peng, Zaihan Jiang, Jun-Hong Cui, "Evaluating selective ARQ and slotted handshake based access in real world underwater networks", In *Proceedings of Wireless Algorithms, Systems, and Applications (WASA)*, 2013, 206–220

Yu Luo, **Lina Pu**, Zheng Peng, Zhong Zhou, and Jun-Hong Cui, "CT-MAC: a MAC protocol for underwater MIMO based network uplink communications," In *Proceedings of the ACM International Conference on Underwater Networks and Systems (WUWNet)*, pp. 1–6, ACM, 2012.

Yibo Zhu, Son Le, **Lina Pu**, Xiaoyan Lu, Zheng Peng, and Jun-Hong Cui, "Aqua-Net Mate: a Real-time Virtual Channel/Modem Simulator for Aqua-Net", In *Proceedings of MTS/IEEE OCEANS*, 2013

Zhao, Yanxiao, Jems Pradhan, Jun Huang, Yu Luo, and **Lina Pu**, "Joint energy-and-bandwidth spectrum sensing with GNU radio and USRP," In *Proceedings of ACM SIGAPP Applied Computing Review*, vol. 2, no. 4, pp. 40–49, 2015.

Zheng Peng, Son Le, Michael Zuba, Haining Mo, Yibo Zhu, **Lina Pu**, Jun Liu and Jun-Hong Cui, "Aqua-TUNE: A Testbed for Underwater Networks", In Proceedings of MTS/IEEE OCEANS, Santander, Spain, June 2011

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ACK	acknowledgement
AGC	automatic gain control
ARQ	automatic repeat request
ATS	available-to-send
CP	cyclic prefix
CTS	clear-to-send
FERI MAC	traffic estimation based receiver initiated MAC
ID	identification
KDC	key distribution center
KLT	Karhunen-Loève decorrelation transformation
LIS	long island sound
MAC	media access control
MBWA	mobile broadband wireless access
MCU	micro controller unit
OFDM	orthogonal frequency-division multiplexing
PMAC	pipelined transmission MAC
PN	pseudo-random noise
RF	radio frequency
RI	receiver-initiated
RIPT	receiver initiated packet train

RTR	request-to-receive
RTS	request-to-send
SASHA	selective ARQ with slotted handshaking-based access
SNR	signal-to-noise ratio
TWN	terrestrial wireless networks
UAN	underwater acoustic network

Chapter 1

Overview

In the recent decades, underwater acoustic network (UAN) has emerged as an active research area due to its wide range of applications from scientific exploration to coastline protection [1–7]. There are significant differences between UAN and terrestrial wireless networks (TWN), due to the unique features of the underwater channel and acoustic modems. Therefore, much of the research work dedicated to TWNs cannot be directly applied to UANs.

To date, significant efforts have been devoted to UAN study to overcome the negative effects introduced by the harsh underwater environments. New transport layer protocols [8,9] have been designed tackling the high latency and sever energy limitation problem in underwater environments. A number of geographical based routing protocols [10–14] are tailored for UANs utilizing the location or depth information in the network. Numerous medium access control mechanisms [15–29] have been proposed for different application scenarios. Localization and synchronization protocols [30–32] in UANs address the challenges from mobile underwater networks with long propagation delays.

Even though extensive studies have been conducted on UANs, most of the studies are based on theoretical analysis or simulation results — little work has been conducted to test algorithms and protocols in real sea experiments.

Most of existing UAN simulators [33–36] have their limitations. For example, it is still challenging to accurately model the underwater acoustic channel. In addition, the grand challenges from underwater environments include but not limited to the long propagation delay and the limited bandwidth. There is still much to explore regarding the real system features and their implications in UANs.

Among the research topics in UANs, the media access control (MAC), which allows users to share the broadcast channel efficiently, the time synchronization, which provides common time reference between all devices in the network, and the cryptographic key technique, which protects network users against threats of eavesdropping and fake data injection, are all critical components in the network stack. In order to provide an insight into the practical UAN design, a number of sea experiments have been conducted to study how UANs behave in the real world. In this dissertation, I explore real system features and their implications in underwater media access control (MAC), time synchronization and cryptographic key generations.

1.1 Introduction

This section provides an introduction on the system features of UANs and a brief overview of underwater MAC, time synchronization and RSS key generation approaches.

1.1.1 Real system features of UANs

Due to the conductivity of water in the nature state, radio signal has high attenuation and cannot propagate far in water. Alternatively, acoustic signal becomes a viable solution for long-distance underwater communications. Despite the advantages

of acoustic signal over radios on communication range, underwater acoustic channels feature some unique characteristics, that in turn, lead to significant challenges on UAN.

The long propagation delay and narrow communication bandwidth are two well-know features and have been extensively explored in the protocol design for UANs. The propagation speed of acoustic signals in water is about 1.5×10^3 m/s, five orders of magnitude lower than the radio propagation speed (3×10^8 m/s). The propagation delays underwater communications are generally in the order of seconds, which make UANs high latency networks. Due to the frequency-dependent attenuation of sound signals, the available bandwidth for underwater acoustic communications is very narrow, usually from tens of kHz to hundreds of kHz. A reasonable transmission rate for existing acoustic modems is tens of kbps when the communication range is around one kilometers.

The grand challenges facing in UAN design include but not limited to those well know facts. There is still much to explore regarding the real system features and their impact on practical UAN design. Underwater testbed design [37–39] and experimental study on UANs thus are highly demanded. The authors in [40] advocated phase-coherent based communication scheme for underwater communications. The performance of the proposed approach was tested at the coast of California, New England Continental Shelf, and Buzzards Bay. The authors in [41] proposed a passive phase conjugation method for underwater acoustic communications to quickly estimate the multipath propagation of the underwater channel in real time. Field experiments have been conducted in Puget Sound near Seattle to test the performance of this method. In [42], the authors investigated the possibility of orthogonal frequency-division multiplexing (OFDM) modulation scheme for high data rate underwater communications and evaluated the performance in two shallow water experiments near Woods Hole,

MA. In their work, the impact of non-uniform Doppler distortion of underwater channel on OFDM modulation scheme was evaluated. However, the works mentioned above mainly focus on point-to-point communications. Only a few tests have been done on the network level [43–46].

1.1.2 Underwater MAC

Underwater MAC is a core module of the UAN systems, which allows multiple users to share the broadcast channel efficiently. Based on the different data transmission mechanisms, underwater MAC protocols can be generally classified into three categories, namely, the random access, handshaking based and scheduling based MAC protocols. Protocols in different categories have distinctive performance on throughput, delay or energy efficiency.

1) ***Random channel access****:* The main feature of random channel access is that senders transmit packets randomly or after a simplified (one-way) contention.

UW-Aloha [15] is a representative random access MAC protocol tailored for UANs. The new features, like the automatic repeat request (ARQ) and back-off scheme are employed to improve the performance of classic Aloha in underwater environments. The authors in [16] proposed two enhanced schemes for Aloha in UANs. In Aloha collision avoidance (Aloha-CA) protocol, each packet is segmented into two distinct parts: a header segment and a data segment. The performance of this scheme improves in that a sensor node can extract the sender-receiver information through a short overhearing. Aloha with advanced notification (Aloha-AN) utilizes an advanced notification (NTF) packet to inform the surrounding nodes of the following data transmission. All nodes maintain a table monitoring the busy durations of each neighboring nodes by overhearing NTF packets. A similar approach called T-Lohi is proposed in [17], which uses a

tone-based contention mechanism to detect collisions. To make T-Lohi work, the tone signal is assumed short enough to eliminate collisions.

2) ***Handshaking based underwater MAC:*** The protocols in this category utilize two-way (or more) handshake to negotiate the channel access. Their motivation is to mitigate the hidden terminal and exposed terminal problems. Handshaking based MAC protocols use small size of control packets to contend and reserve channel for data transmissions, in which collisions can be avoided with a proper design.

A typical handshaking MAC is Slotted FAMA [18], using request-to-send (RTS) and clear-to-send (CTS) to reserve time slots for data transmissions. Distance-Aware Collision Avoidance Protocol (DACAP) [19] allows the sending of CTS to be fast to reduce the waiting time at the sender, hence increases the throughput of the network. Adaptive propagation-delay-tolerant collision-avoidance protocol (APCAP) [20] and Reservation-based MAC protocol (R-MAC) [21] utilize the interval time between handshaking signals to process other packets such that the channel utilization is improved. COPE-MAC [22] further improves the channel efficiency by using a parallel reservation and cyber carrier sensing scheme. Noh et al. proposed the Delay-aware Opportunistic Transmission Scheduling (DOTS) method [23] which uses topology information of the network and handshaking mechanism to improve the performance of the protocol.

3) ***Scheduling based underwater MAC:*** The scheduling based MAC protocols tend to preassign the time and/or frequency resources to nodes in a network [47]. The classical scheduling MAC, such as TDMA, FDMA and CDMA have drawbacks of poor performance in terms of throughput and latency in networks with dynamic traffic loads.

In order to solve the spatial-temporal uncertainty problem [45, 46, 48] in the transmission scheduling, a couple of effective spatial or time reuse MAC protocols have been proposed. UWAN-MAC [25] leverages local synchronization to arrange the time

line of each node for energy efficiency improvement, but the requirement of ultra low traffic rates in a network constraints its applications. Hybrid spatial reuse TDMA (HSR-TDMA) [26] utilizes a graph coloring algorithm to improve channel utility efficiency, but does not well address the hidden and exposed terminal problem in multi-hop UANs. Spatial-Temporal confliction graph is applied in Spatial-Temporal MAC scheduling (ST-MAC) [49] to handle the spatial-temporal uncertainty problem.

1.1.3 Underwater time synchronization

Due to the application requirements and the severe resource constraints in an underwater environment, most of UANs are designed as distributed system. The time synchronization, which provides common time reference between all devices in the network, is widely demanded by applications including sensor data collection [50], network localization [51] and coordination in MAC and cooperative communications [52, 53].

Time synchronization has been a research area of long history [54]. Many synchronization protocols have been proposed and tested in the TWN [55–59]. The source of delays and uncertainties in message delivery have been extensively studied in the radio system. The effect of these uncertainties on synchronization protocols have been evaluated and schemes to improve precision have been proposed [56, 57]. However, the environments of the oceans and grounds are very different, leading to distinct designs between acoustic and radio systems. Due to the unique features of UANs, the territorial time synchronization protocols may need an overhaul before using them efficiently in underwater environments.

Recently, the time synchronization for UANs has drawn people's attention along with the development of underwater communication and networking technology. Synchronization protocols, such as [31, 60, 61], have been proposed for the high latency and mobile underwater networks. In these works, the long propagation delay of the

acoustic signal and the mobility of the acoustic nodes are carefully studied, and the errors caused by these unique features are well compensated.

1.1.4 Underwater secure communication

Like terrstrial sensor networks, UANs are susceptible to various attacks, which target different components in the system. For example, attacks like wormhole target at routing protocols [62], and jamming attacks can disrupt links between nodes [63–65]. An adversary can also violate communication security by passively eavesdropping the private signal or actively injecting fake information to the network [66]. Among the aforementioned security issues, the communication security is one of the most fundamental and critical tasks in underwater networks, which use broadcast channel for acoustic transmissions. The cryptographic key technique aims to protect network users against threats of eavesdropping and fake data injection. The public-key cryptography are nearly infeasible in the networks with constrained energy and processing power [67]. Alternatively, symmetric-key ciphers are often used to provide confidentiality in underwater communications because of their performance advantages [68, 69].

However, symmetric-key cryptography require a shared secret key by sender and receiver for both encryption and decryption. The requirement that both parties have access the the secret key makes the key generation and key exchange challenging, especially in resource constrained UANs. It is difficult, if not impossible, to specify an online key distribution center (KDC) in oceans to allocate secrete keys among devices. The most accepted solution is a combination of pseudorandom key generators and key predistribution [68, 69]. However, lack of randomness in those generators is a common problem leading to cryptanalytic breaks. Key predistribution have connectivity and resiliency issues. An isolated node possibly exists when it has no common key with neighboring nodes. All the methods that preinstall keys on the nodes also have the

risk that a single compromised node might result in a number of unsafe parties sharing common keys with the compromised node.

RSS-based key generation schemes [70–73], however, allows each pair of nodes, after being deployed, to update secret keys easily at any time. In RSS-based key generation schemes, the randomness of the keys depends on the entropy naturally available in the environments. The communicating parties on the two ends of a reciprocal link can produce a shared key through local RSS measurements [74, 75]. An opponent that is monitoring the communication channel, however, can hardly guess the secret key if it is physically near neither communicating entities [76]. This security is consequently ensured with the spatial diversity of acoustic channel, as shown in Fig. 1.1. Therefore, the RSS-based methods become promising techniques for underwater secure communications.

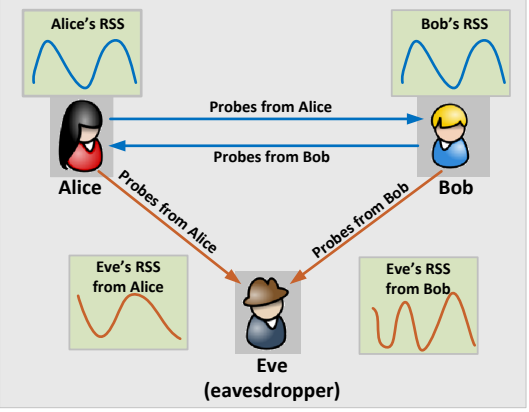


Figure 1.1: RSS-based key generation in the network.

1.2 Contributions of this dissertation

Following this direction,a number of real sea tests have been conducted from 2011 to 2014. According to the experiment results, I observe several phenomena that have significant impact on the performance of UANs, including **(1)** long preamble of acoustic modems, **(2)** heterogeneous packet delivery, **(3)** spatial and temporal communication

range uncertainty, **(4)** multi-hop interference and **(5)** delayed data transmission. The impact of aforementioned phenomena on each layer in the network stack are discussed in this dissertation work.

In the underwater MAC perspective, I provide a realistic and quantitative comparison of the performance of a variety of underwater MAC protocols: random access based UW-Aloha [15], handshaking based SASHA (selective ARQ with slotted handshaking-based access) [24] and scheduling based PMAC (pipelined transmission MAC) [47], in multi-hop networks [43, 45, 46]. A comparison of protocols is conducted on system performance such as throughput, delay, deliver ratio and energy efficiency. Based on the field observations and result analysis, I discuss the impacts of the real system features on different MAC mechanisms and propose a traffic estimation based receiver initiated MAC, called FERI MAC. In FERI MAC, receivers replace the role of senders in conventional MAC protocols as the initializer of a handshake process. It allows receiver to establish handshake with multiple senders in parallel, improving the network performance in terms of throughput, delivery delay and energy efficiency.

On time synchronization in UANs, I identify the source of delays in message deliveries based on the system feature of real acoustic modems [77]. The delivery delay is decomposed into seven parts, namely, the command generation time, command delivery time, command detection time, transmission preparation time, message transmission time, signal propagation time and message reception time. The magnitude of each delay and its statistics are measured in this work. I evaluate the performance of representative sender-receiver based TSHL-RS and receiver-receiver based RBS-UW in real systems. Discussions on the performance improvement on time synchronization are also provided based upon the experiment results.

On RSS key generation in UANs, the RSS based key generation methods are explored and evaluated in sea trials. More specifically, I discuss the advantages of RSS

key generation and analyze the grand challenges from the unique features of UANs. Furthermore, I conduct sea tests and evaluate the performance of three representative RSS based key generation approaches, namely Ano, Mathur and Patwari, for underwater secure communications [78]. From sea experiment, I explore how underwater system features affects the representative RSS based key generation protocols and provide solutions to improve the performance in terms of key generation rate, randomness and key agreement probability.

1.3 Dissertation roadmap

The remainder of this dissertation is organized as follows. Firstly, I introduce my new findings on the real system features in UANs in Chapter 2. Then, I evaluate the performance of representative underwater MAC protocols through sea experiment results in Chapter 3 and propose a practical underwater MAC design. In Chapter 4, I present the experiment results of TSHL-RS and RLS-UW and provide an insight into practical time synchronization protocol design and performance improvement. In Chapter 5, I investigate and evaluate the performance of representative RSS based key generation approaches in sea trials. Solutions to improve the performance of RSS based key generation in UANs are provided. At last, I conclude my contributions in Chapter 6.

Chapter 2

Real System Features of UANs

In underwater environments, the radio signal suffers from heavy attenuation and thus cannot propagate far. Alternatively, acoustic communication becomes available solution. However, this different communication method poses great challenges to underwater wireless networks. Compared with territorial wireless networks, acoustic signals in water propagate much slower, with a speed about 1.5×10^3 m/s, five orders of magnitude lower than the radio in air (3×10^8 m/s). Therefore, the low sound speed introduces long propagation delay issue in UANs, which has been covered in current UAN design. Furthermore, the bandwidth capacity of underwater acoustic channels is very limited and heavily depends on both the transmission range and the frequency [79]. According to [80], nearly no research or commercial modem can exceed 40 km $\times$ kbps as the maximum attainable range-rate product.

Although the long propagation delay and the limited available communication bandwidth have been fully investigated in the recent literature [17–20], some problems caused by the characteristics of real acoustic modems, such as the long preamble sequence, are still overlooked in the protocol design and performance analysis. In order to explore more system features of UANs, a series of real sea experiments were conducted from 2011 to 2014. In this chapter, I introduce several phenomena revealed in

sea tests, including **(1)** long preamble of acoustic modems, **(2)** high packet loss ratio and heterogeneous packet delivery, **(3)** spatial and temporal communication range uncertainty, **(4)** multi-hop interference and **(5)** delayed data transmission caused by the clock skew and the busy terminal problem of acoustic modems. The content in this chapter is mainly based on my previous work published in [43][1] and [45][2] .

2.1 Experiment settings

In order to identify the unique features of acoustic system, a series of sea experiments were conducted from 2011 to 2014.

In May 2011, we did field test at Chesapeake Bay, Maryland. Eight nodes were deployed in a ring topology with about 1 km average distance between neighbor nodes, as shown in Fig. 2.1. Each node in the network had a Teledyne Benthos ATM-885 modem [81] connected to a surface buoy, which was equipped with a Gumstix Verdex Pro XM4 single-board computer. The Benthos modems operated at 800 bps for the purpose of reliable transfer.

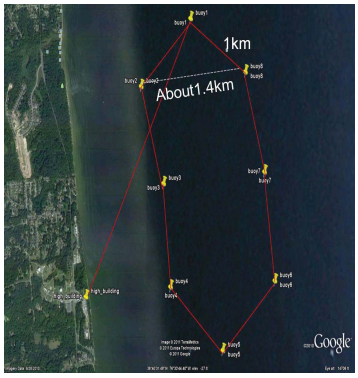


Figure 2.1: Experiment in Chesapeake Bay, Maryland, in May 2011.

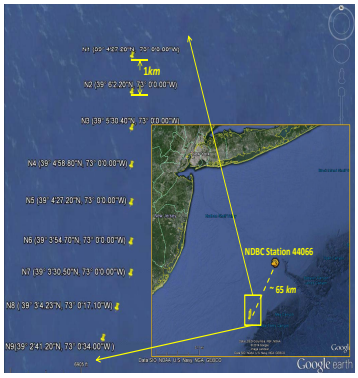


Figure 2.2: Experiment in Atlantic Ocean, in September, 2012.

Figure 2.3: Experiment in LIS during November, 2013–April, 2014.

[1] © 2015 IEEE. Reprinted with permission from Lina Pu, Impact of real modem characteristics on practical underwater MAC design, IEEE OCEANS, May. 2012

[2] Reprinted from Computer Communications, Vol. 56, Lina Pu, etc., Comparing underwater MAC protocols in real sea experiment, Pages 47 – 59, Copyright (2015), with permission from Elsevier

In September 2012, we conducted another sea experiment in the Atlantic Ocean using Teledyne Benthos modems. Nine underwater acoustic nodes were deployed as a string topology, the coordinates of which are shown in Fig. 2.2. The strip area of this experiment was about 120 km off New Jersey shore with average water depth of 80 m. The acoustic modems were deployed about 30 m below the sea surface. In this deployment, the average distance between two neighboring nodes was about 1 km, as shown in Fig. 2.2. The horizontal distance between two end-nodes in the network was about 7.3 km.

During the period from November 2013 to April 2014, we had three nodes at Long island sound (LIS) as shown in Fig. 2.3. The UConn OFDM modem was equipped on each node for acoustic communication. The center frequency and the bandwidth of the acoustic modem are 17 kilohertz and 6 kilohertz, respectively. All modems were deployed about 20 meters below the surface, and water depth of around the experiment site was about 30 meters.

2.2 Real system features of UANs

From the experiment results, I revealed several problems that have never been well studied before, such as (1) long preamble of acoustic modems, (2) high loss ratio and heterogeneous packet delivery, (3) spatial and temporal communication range uncertainty, (4) multi-hop interference and (5) delayed packet transmissions. In the rest of this section, I introduce five observations in details with experiment results.

2.2.1 Long preamble of acoustic modems

In acoustic modems, for the purpose of synchronization, burst data sequence detection, automatic gain control (AGC) as well as channel response estimation, a preamble must be designed as a prefixing of each packet in acoustic modems [82–86]. In radio

Table 2.1: Control Packet (6 Bytes) duration with different acoustic modem

Modem Type	Data Rate	Preamble Length (s)	Control Packet Length (s)
Benthos ATM-88X	800bps (Standard)	≈ 1.50	≈ 1.56
	2.4Kbps (Highest)		≈ 1.52
UConn OFDM	3.045Kbps	0.49	0.66
WHOI Micro	80bps (Standard)	0.87	1.47
	300-5000bps (PSK mode)		1.52

systems, the duration of the preamble is very short, normally within several hundreds of microseconds. For instance, the preamble in IEEE 802.20 mobile broadband wireless access (MBWA) standard is constituted of 8 symbols with 104 μs for each symbol (i.e. totally 832 μs) [87]. On the contrary, the preamble in underwater acoustic modems can be up to one second, three orders of magnitude higher than that in radio systems. The length of preambles in representative acoustic modems are listed in Table 2.1.

The long preamble in UANs essentially results from two folds, the low data rate of acoustic modems and the long multipath of underwater channel.

- **The low data rate of acoustic modems** is the primary reason accounted for the long preamble problem in UANs. The synchronization sequence is one crucial part of preamble for packet reception. To achieve a good synchronization performance, a sequence of hundreds of known bits, such as pseudo-random noise (PN) sequence, is usually employed in communication systems. In radio networks, the high data rate insures the duration of this synchronization signal to be very short. However, the low data rate of acoustic modems (listed in Table. 2.1) largely increases the transmission time of the sequence of same length. Taking 512 bits of PN signal as an example, it only lasts 17 μs in IEEE 802.20 standard with 30 Mbps data rate. The transmission time of this PN signal in UANs, however, extends to 0.64 s with 800 bps data rate.

- **The long multipath of underwater channel** is another fact that contributes to the long preamble of acoustic modems. The preamble is usually constituted of several blocks, each block serving for different functionality. To against the inter-block interference in long multipath environments, guard time for PSK and FSK based modems or cyclic prefix (CP) for OFDM based modems are needed between blocks. The length of the guard time or the CP sequence depends on the multipath effects. Radio channel has short multipath time owning to the high propagation speed of the electromagnetic signal. The guard time or the CP thus is as short as tens of microseconds (e.g. 4.7/16.7/53.3 μs in 3GPP LTE standard for different channel conditions). Contrarily, underwater channel suffers from sever multipath in tens of milliseconds or even longer, depending on the deployment and channel condition. The required guard time or the length of CP in such long multipath time environment is thus considerably increased by almost 1000 times than that in radio networks.

The long preamble feature of practical acoustic modems challenges almost all existing UAN protocols, especially for those using small control packets. Take the Benthos modem [82] as an example. Even a packet carring only several bytes of useful information, becomes longer than 1.5 seconds with the fixed preamble sequence. This implies that "short" control packets are not short anymore and will equally suffer heavy collision or channel loss as data packet in the network. Another example is the UConn OFDM Modem [86] - the high speed OFDM modem developed by the UConn UWSN lab. In the UConn OFDM Modem the packet consists of two preamble blocks for packet detection as well as synchronization, and OFDM data blocks for data transmission. Similar to Benthos modem, no matter how few useful data bytes is carried on, the duration time of any packet is no shorter than 0.66 seconds. Fig. 2.4 displays the

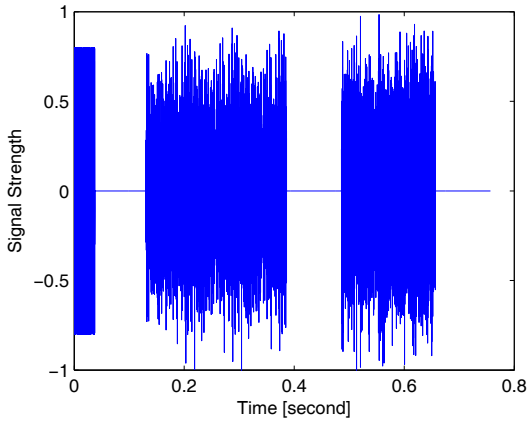


Figure 2.4: The signal structure of UConn OFDM modem. The first two blocks are preamble used for signal detection and synchronization respectively. The data section starts from the third block, with each block of the same size.

minimum packet transmission time with two preamble blocks and one data block in UConn OFDM Modem.

2.2.2 Heterogeneous packet delivery

Due to the geometric spreading of acoustic signals in propagation and the severe absorption in water, the high error rate in underwater communications is a well understood fact to the research community. However, the packet loss ratio revealed in the field results was not only severer than we expected, but also varied significantly through the network.

I calculated the fraction of lost packets on different links and displayed the result in Fig. 2.5. In the test, 9 nodes were deployed in a line as illustrated in Fig. 2.2, among which 7 intermediate nodes (N2 to N8) took turns to send data of 200 Bytes. I record the ratio of packets not received by the receivers at both sides and show loss ratio in Fig. 2.5. This experiment result was an average of a one-hour test. 22.53% packets out of the total transmissions in the network failed to reach the receivers of 1 km away. This was the best reliable Benthos modems could achieve with the concurrent channel

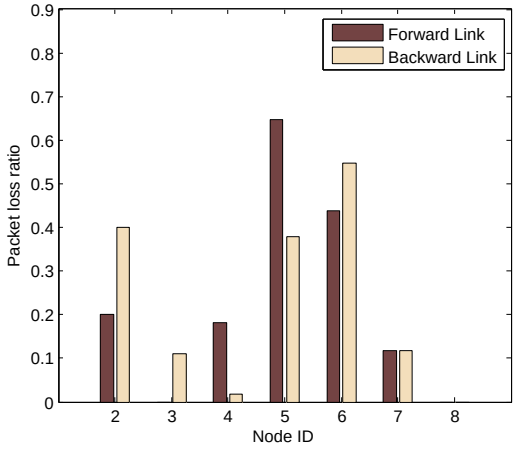


Figure 2.5: Packet loss ratio on different links.

condition, as the modem worked at 300 bps with 25 ms multipath guard time and 1/2 convolutional coding. This result is magnitudes higher than the PER people usually assumed in the simulations.

The loss ratio was not only high but also varied significantly among different links. Due to the heterogenous geometry of sea surface and sedbed, the diversed multipath effect could result in greatly different communication quality through the network. We had observed this phenomenon when we towed the modems around trying to get reliable communications. Good links on N3, N7 and N8 had less than 15% loss ratio in my observation, compared with almost 65% packet loss with bad channels on N5. Here Fig. 2.5 came from a random chosen test. Even though packet loss ratios were not exactly the same among other tests, similar diversity on packet loss ratios was observed.

Beside the substantially varied loss ratio on difference nodes, the packet delivery features heterogeneous when packets travel in different directions. Taking the network in Fig. 2.2 as an example, I call the links for southern directional communication as forward links and the reverse ones as backward links. In Fig. 2.5, the forward links suffered severer packet losses than the backward links when $N4$ and $N5$ were

sending. On the contrary, the forward links for $N2$, $N3$ and $N6$ had better reliability than the reverse links. This significant varied delivery means that packets traveling in different directions can have dramatically different packet loss ratios, and therefore brings troubles to MAC protocols relying on homogeneous network assumption.

The high packet loss ratio and heterogeneous deliveries across the network is a result of the complicated underwater environments. For the acoustic modems like Benthos ATM-885 using FSK based modulation scheme, the severe multipath effect is one of the most fundamental obstacle to robust underwater communications. How to deal with the high loss ratio and diversed packet deliveries among the network will be a big challenge on the practical MAC design in UANs.

2.2.3 Communication range uncertainty

Under a combined impact of the broadcast nature of acoustic signal and the unstable underwater channel condition, the network communication range demonstrates uncertainty both spatially and temporally. Communication range is closely related to the transmission power of acoustic signal, distance, underwater channel quality and local noise level at the receiving point. I am going to investigate both spatial and temporal uncertainty with regard to transmission range.

In this subsection, I define two nodes are connected if packets from the sender can reach a receiver at a ratio no lower than 1/3. Although 1/3 seems to be a low reception ratio, given the high packet loss ratio I observed, it is actually a relatively decent one. Another reason why I choose 1/3 as the threshold is that UW-Aloha allows a maximum of 3 retransmissions for a data packet. Therefore, 1/3 means a data packet can be received by a receiver in UW-Aloha within maximum retransmission attempts.

Table 2.2 lists the successful reception ratios at nodes with different distances when $N1$ to $N9$ sent respectively. I classify the reception ratios into three categories. The

Table 2.2: Packet deliver ratio along the path
(**Bold** - reliable one-hop communication with higher than 1/3 delivery ratio, ***Bold italic*** - reliable multi-hop communication with higher than 1/3 delivery ratio, *Italic* - unreliable one-hop communication with lower than 1/3 delivery ratio).

Sender ID	Successful Reception Ratio								
	N1 (%)	N2 (%)	N3 (%)	N4 (%)	N5 (%)	N6 (%)	N7 (%)	N8 (%)	N9 (%)
N1	–	**40.6**	22.3	6.9	1.7	0.0	0.0	0.0	0.0
N2	**67.0**	–	*20.5*	0.8	1.7	0.0	0.0	0.0	0.0
N3	***60.2***	**92.9**	–	**56.1**	25.5	0.0	0.0	13.3	6.1
N4	20.4	32.6	**80.1**	–	**50.3**	5.0	0.6	***45.9***	***48.6***
N5	6.0	16.5	14.7	**66.2**	–	*28.1*	7.2	***66.2***	***44.3***
N6	0.0	0.0	4.6	2.3	**35.6**	–	**56.8**	***47.7***	12.1
N7	0.0	0.0	0.0	0.0	6.0	**67.2**	–	**85.1**	16.4
N8	0.0	0.0	2.0	2.0	2.0	2.0	**81.6**	–	**100.0**
N9	0.0	0.0	8.3	22.9	16.7	14.6	22.9	**100.0**	–

bold ones are the reception ratios no lower than 1/3 between two adjacent nodes, which means a node gets at least one hop communication range. The ***bold italic*** ones are the reception ratios no lower than 1/3 between two nodes which are two hops or further away. This indicates a node can receive data packets reliably from a node further away than one hop. In another word, the **bold** ones represent a reasonable reliability while the ***bold italic*** ones stand for an over high reliability. The *italic* ones are the reception ratios lower than 1/3 between two adjacent nodes, which means a low reliability. Based on this reception ratio table, we can observe the communication range of each individual node between $N1$ and $N9$.

Spatial communication range uncertainty emerges in two aspects. (1) The communication ranges of different nodes varied significantly. As shown in Table 2.2, nodes $N1$, $N2$ and $N7$ to $N9$ were only able to communicate to neighbors within one-hop away at reception ratios higher than 1/3, which means the communication range was only one hop. However, $N5$ was able to reach $N9$ with 44.3% reception ratio, indicating

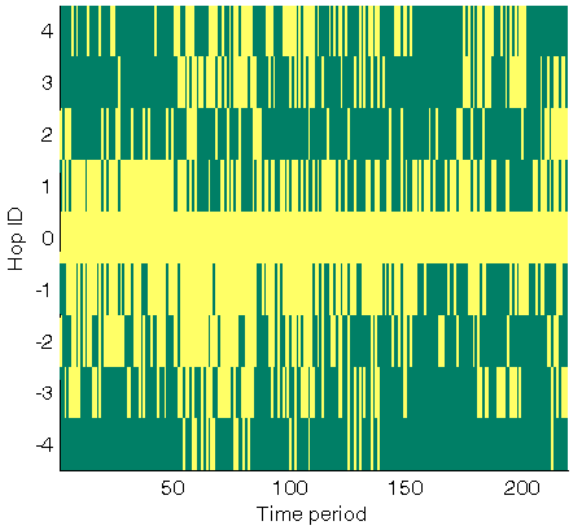


Figure 2.6: Transmission range changes with time.

that it had a maximum of 4-hop coverage. Even if all nodes operate at the same power, frequency band and data rate, the network has heterogenous communication reliability, which challenges the MAC design and protocol evaluation for UANs. (2) When one sender transmitted, receivers at different locations had evidentially variant reliability. There exists a possibility that closer receivers have much worse communication than the nodes further away, which was observed when $N4$ and $N5$ transmitted.

In the UAN protocol design, it usually assumes uniform communication range. When the positions of nodes are determined in the deployment, the network topology is considered to be fixed and pre-known for protocol evaluation. The existence of spatial transmission range uncertainty, however, leads to unexpected severer collisions in the region with larger coverage than people assume. The occasional high loss ratio on weak links in the network will also degrade the performance of UAN protocols. When the actual topology in real applications does not match the topology used in the analysis, it will cause non-negligible gap between theoretical and experimental results.

The communication range not only varies spatially, but also shows dynamic nature in temporal dimension. Owing to the time varying nature of wind, current, marine

mammal noise and man made activities, the link reliability feature changes with time. Fig. 2.6 illustrates the dynamic communication range when the middle node (N5) sent packets of 200 bytes. The x-axis on the graph is the index of transmitted packets ordered by sending time, which stands for the time records. The y-axis gives the transmission range in the unit of hops. The positive and negative Hop IDs represent the transmission on the two directions. Packets were sent from Hop 0 in this figure. I highlight the region where nodes successfully received packets from the sender. As shown in Fig. 2.6, the transmission range has remarkable variation with time. In some time periods, no packets could be reliably delivered to any node further than one hop away. On the contrary, in the rest of time the sender had good communication reliability for transmissions on both directions.

Moreover, Fig. 2.6 illustrates the non-uniform packet deliveries across the network. Even when the packet reached nodes four hops away, huge packet losses happened on the intermediate nodes on the same direction. This indicates that it becomes inappropriate to use a predetermined radius to represent the transmission range of acoustic communications. Within the transmission range, it is conventionally assumed to have uniform transmission reliability or to have monotonous increased loss ratio with distance. However what happens in the real ocean environment is that a node may have reliable communications with nodes further away while failing to reach closer nodes. This poses challenges to underwater routing and topology management.

2.2.4 Multi-hop interference

As a result of spatially and temporally communication range uncertainty, signals originated from multiple hops away are hard to eliminate in the network. The multi-hop interference becomes a critical issue for large scale UANs. I study the relationship between packet loss ratio and the strength of interference via experiment results.

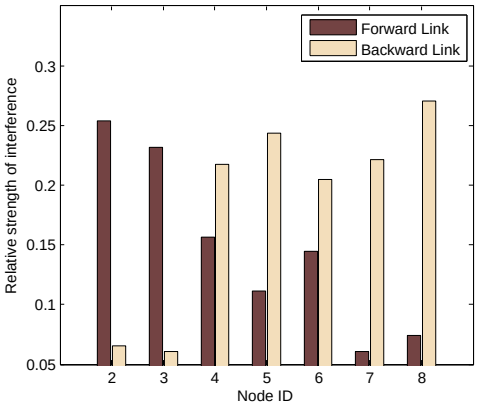


Figure 2.7: A metric related to the strength of interference.

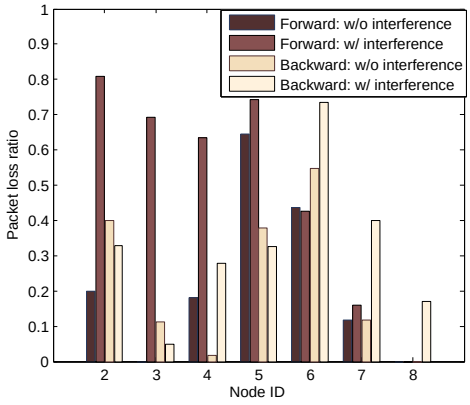


Figure 2.8: Packet delivery ratio with respect to the strength of interference.

The packet loss ratios on most links in the network were lower than 20% when the communication was not interfered by any other transmissions, as shown in Fig. 2.8. On the assumption that the communication range of acoustic modems was one hop, two nodes with a distance equal to three hops away, e.g. $N1$ and $N4$, had a chance to transmit simultaneously. When the signal reached a node further than one hop away with high energy, the signal would become an interference and corrupt the intended reception on the node. I call this phenomenon as multi-hop interference. The significant increment on packet loss ratio revealed in Fig. 2.8 indicates the strong negative effect of multi-hop interference in large scale UANs.

However, I do observe comparable deliveries between the scenarios with and without multi-hop interference for the forward link on $N6$ to $N8$ and the backward links on $N2$, $N3$ and $N5$. In order to quantify the strength of interference, I measured the inverse of distance square $(1/D^2)$ to represent the relative strength of interference, since spreading loss dominates over the frequency dependent absorption at short range acoustic communications.

Fig. 2.7 shows the average strength of interference on forward and backward links for the experiment in Fig. 2.8. On the forward links of $N1$, $N2$ and $N3$ and backward

links of $N4$, $N6$ and $N7$ where the interference had strong effects, the relative strength of interference was higher than 0.15, which means the interference signal had 85% higher spreading loss than the signal of intended reception. The packet loss ratio was hardly affected for links on both directions when the relative strength of interference was lower than 0.15, as shown in Fig. 2.8. This indicates 0.15 a strong strength for the interference from multiple hops away to corrupt intended reception in my experiment.

In underwater networks, the communication reliability is hardly predictable not only for heterogeneous acoustic channel quality, but also because of the varied strength of multi-hop interference. Signals from distant nodes might have significant impact on the packet delivery, which challenges the acoustic communications and medium access control in underwater networks.

2.2.5 Delayed data transmission

In the network stack, both MAC and synchronization are sensitive to the delays in data transmission. The purpose of MAC is to handle the interference in a shared medium by precisely transmission scheduling. Time synchronization protocols are generally implemented on the upper layer and utilize the MAC layer time stamping for clock synchronization. Any unexpected delays incurred in packet transmission will be finally added up to the synchronization errors. In the literature, people usually assume zero or negligible delay between the actual packet transmission time on acoustic modems and the scheduled sending time on MAC layer. However, in the field experiments, I discovered considerable time difference between the scheduled transmission time and the actual modem sending time. The major reason for extra delays might be the clock drift and the busy terminal problem of acoustic modems [43].

The real time scheduling of Aqua-Net was paced by the CPU time of the Gumstix microcomputer. Before each test, 9 nodes were manually synchronized with the satellite

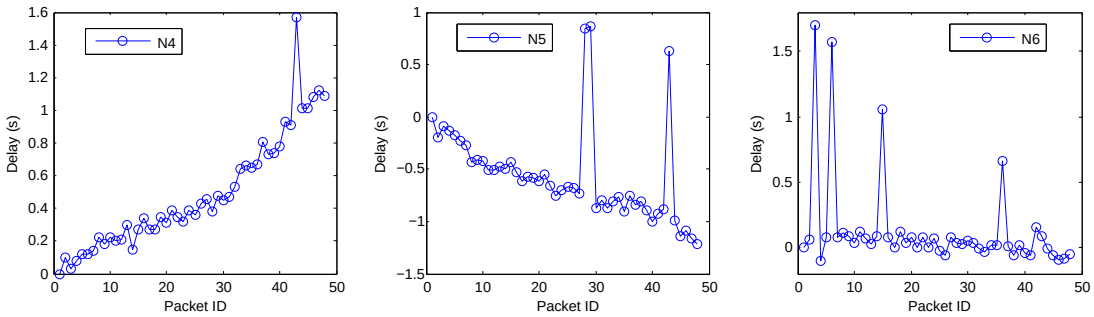


Figure 2.9: Delays before modem transmission.

server. No time synchronization was implemented within each test. I calculate the interval between modem sending time and MAC scheduling time and display the delays in Fig. 2.9.

Due to the clock skew of Gumstix systems, slightly delayed transmission or sending ahead of scheduling were observed on $N4$ and $N5$ in Fig.2.9. At the end of three hours test, a 1.2 seconds delay was accumulated on $N4$, owing to a faster clock on $N5$ than that of on $N4$. Similarly, shorter delays occurred on $N5$ when it communicated with $N6$, which ran normally. Even though no obvious clock skew had been observed when Gumstix ran off-loaded before experiments, the time drift was not a rare issue in the real test. Across the 9 nodes in the sea experiment, two Gumstix controllers were evidentally slower or faster than the rest seven. On-board time synchronization becomes a critical component in the real system, especially when the test spans a long time.

Besides the time drift of system clock causing the skewed packet transmission, the busy terminal problem of acoustic modems introduced impulsive delays. In current acoustic modem design, the actions of packet transmission and reception cannot be interrupted once they get started. This implies that the modem has to receive the whole packet forcibly without dropping out halfway, even when this packet is an overheard packet destined for other nodes. This phenomenon is called busy terminal problem. If

the modem is busy with either sending, receiving or overhearing at a scheduled packet transmission time, the outgoing packet pushed into the modem can not be processed immediately as the MAC protocol has scheduled. The packet transmission will be postponed until the modem comes out of busy state. This busy terminal problem is aggravated by the long transmission time in acoustic networks. In my experiments, the packet of 200 bytes lasted for 7.4 seconds when Benthos modems operated at 300 bps.

When the packet transmissions were not interfered by the busy terminal problem, the modem sending delays were randomly distributed from 0 to 200 ms. Among 48 packet receptions on N6 in Fig. 2.9, four significant delays were observed in one test. This means the delayed modem transmission was not a rare situation. The highest delay was up to 1.8 seconds, which implies that the actual data sending time was 1.8 seconds later than the transmission scheduled by the MAC protocol. These four modem sending delays were considerable even compared with the 0.7 seconds propagation delay and could make the collision avoidance mechanism futile. The delays introduced by the busy terminal problem are impulsive and unpredictable events and therefore pose grand challenges to the protocol design for underwater MAC and time synchronization.

2.3 Summary

To summarize, I observed the long preamble of acoustic modems, high packet loss ratio and heterogeneous packet deliveries, spatial and temporal communication range uncertainty, multi-hop interference and delayed data transmission on acoustic modems, from real field experiments.

The long preamble feature challenges almost all protocols including MAC, routing, synchronization and localization, especially for those with small control packets. Dealing with high packet loss ratios becomes a challenging task for MAC protocol design, as

packet losses could dramatically degrade the MAC performance from all aspects. Collision avoidance handling, transmission scheduling and MAC performance analyzing rely on topology information which suffers a lot uncertainties on the communication range in both spacial and temporal dimensions. The multi-hop interference from distant senders in the networks further degrades the communication reliability in a remarkable way. The diversified strength of interference become another challenge faced by MAC protocols assuming homogenous network reliability. Unexpected sending delays on acoustic modems are critical due to the system clock drift and the busy terminal problem. Especially in dense networks where nodes experience heavy overhearing, impulsive packet transmission delays would cause significant collision avoidance failures. Long term applications of UANs call for efficient time synchronization mechanisms considering the clock skew in field experiments. In the following chapters, I discuss the impact of aforementioned features on underwater MAC, synchronization and security protocols in details.

Chapter 3

Impact on Underwater MAC

Underwater MAC, which allows users in the network to share the channel efficiently, has been the most critical part among the research topics in UANs. Due to the unique features of the underwater channel and acoustic modems, much of the research work dedicated to TWNs cannot be directly applied to UANs.

To date, significant efforts have been devoted to the underwater MAC protocol design to overcome the negative effects introduced by the harsh underwater environments. The authors in [16] studied Aloha based protocols in single hop underwater networks. An adaptive propagation delay tolerant collision avoidance protocol (APCAP) was designed in [20] to mitigate the long propagation delay problem in UANs. To solve the same problem, the authors in [27] and [28, 29] designed scheduling based ST-MAC and CT-MAC respectively, both of which give a comprehensive performance evaluation via simulations. Although some well-known features of UANs including the long propagation delay and the low data rate [79] have been taken into account, the performance of these protocols and the conclusions drawn in these works have not been verified in real experiments.

Most of existing UAN simulators [33–36] have their limitations. For example, it is still challenging to accurately model the underwater acoustic channel. In addition, some

unique features of the UAN can only be revealed in real world sea tests. Therefore, there have been some works on testbed design and experimental study on underwater acoustic communications [15, 37]. For instance, the authors in [40] advocated phase-coherent based communication scheme for underwater communications. The performance of the proposed approach was tested at the coast of California, New England Continental Shelf, and Buzzards Bay. The authors in [41] proposed a passive phase conjugation method for underwater acoustic communications to quickly estimate the multipath propagation of the underwater channel in real time. Field experiments have been conducted in Puget Sound near Seattle to test the performance of this method. In [42], the authors investigated the possibility of OFDM modulation scheme for high data rate underwater communications and evaluated the performance in two shallow water experiments near Woods Hole, MA. In their work, the impact of non-uniform Doppler distortion of underwater channel on OFDM modulation scheme was evaluated.

However, the works mentioned above mainly focus on point-to-point communications. Only a few tests have been done in the network level [43, 44]. The work in [43] has uncovered the significant preamble length in acoustic modems and its adverse effect to MAC protocols with small control packets. In [44], the authors conducted sea tests to evaluate the performance of three MAC protocols including CSMA, T-Lohi and DACAP in terms of throughput, efficiency and packet latency in varied single-hop scenarios and a two-hop network. The authors in [45, 46] revealed several system features of UANs, such as heterogeneous packet delivery, transmission range uncertainty, multi-hop interference and delayed data transmissions, through sea experiments. The authors also studied the impact of those practical issues on the MAC design for real multi-hop networks. These works have exposed some important facts which can only be observed in the field tests. They give researchers some valuable information of the underwater MAC performance in the real world environments.

The grand challenges facing in underwater MAC protocol design include but not limited to the long propagation delay, the limited bandwidth and the long preamble in acoustic modems. There is still much to explore regarding the real system features and their impact on MAC performance. Following this direction, in this chapter, I evaluate different underwater MAC protocols in sea tests. Through experiment results, I further analyze the impact of the real system features, which are introduced in Chapter 2, on practical MAC design. The content in this chapter is mainly based on my previous work published in [45][1] and [88][2] .

3.1 Background and related work

Underwater MAC is a core module of the UAN systems, which allows multiple users to communicate through a shared medium. Essentially, a MAC protocol is designed to avoid collisions among network nodes by properly scheduling when nodes transmit. The MAC protocols have significant impact on the network performance in terms throughput, delay, delivery ratio and energy efficiency. Different underwater applications have varied demands on MAC protocols. Delay sensitive UANs requires high throughput and short end-to-end delay, while energy efficiency might be unconsidered. In a long term UAN application, like ocean monitoring, the high energy efficiency is usually a top priority. The UANs with heavy traffic loads would have high requirement on network throughput. A practical and efficient MAC protocol is critical to make UANs feasible in the real world.

Based on different transmission mechanisms, underwater MAC protocols can be generally classified into three categories, namely, random access, handshaking based and scheduling based protocols.

[1] Reprinted from Computer Communications, Vol. 56, Lina Pu, etc., Comparing underwater MAC protocols in real sea experiment, Pages 47 – 59, Copyright (2015), with permission from Elsevier

[2] © 2013 ACM. Reprinted with permission from Lina Pu, Traffic estimation based receiver initiated MAC for underwater acoustic networks, ACM WUWNet, Nov. 2013, doi: 10.1145/2532378.2532412

3.1.1 Random access based underwater MAC

Random channel access is the main feature of Aloha based MAC protocols, where senders transmit packets randomly or after a simplified (one-way) contention.

UW-Aloha [15] is a representative random access MAC protocol tailored for UANs. The new features, like the ARQ and back-off scheme are employed to improve the performance of classic Aloha in underwater environments. The authors in [16] proposed two enhanced schemes for Aloha in UANs. In Aloha collision avoidance (Aloha-CA) protocol, each packet is segmented into two distinct parts: a header segment and a data segment. The performance of this scheme improves in that a sensor node can extract the sender-receiver information through a short overhearing. Aloha with advanced notification (Aloha-AN) utilizes an advanced notification (NTF) packet to inform the surrounding nodes of the following data transmission. All nodes maintain a table monitoring the busy durations of each neighboring nodes by overhearing NTF packets. A similar approach called T-Lohi is proposed in [17], which uses a tone-based contention mechanism to detect collisions. To make T-Lohi work, the tone signal is assumed short enough to eliminate collisions.

The Teledyne Benthos ATM-885 modems used in the experiment support neither separate decoding on the header segment nor tone signal transmissions. For this reason, we tested UW-Aloha as a representative random access MAC protocol. Fig. 3.1 shows the work flow of UW-Aloha. A sender randomly transmits packets without a reservation or a negotiation with other senders. Due to the long propagation delay in underwater environments, collisions cannot be sensed immediately by listening to the channel as what can be done in TWNs. Therefore, UW-Aloha incorporates acknowledgement (ACK) to explicitly informs the sender whether the transmission is successful or not. If a sender cannot receive an ACK in time, it means either the data or the ACK is lost due to collisions or link errors. The sender will back-off before retransmitting the data

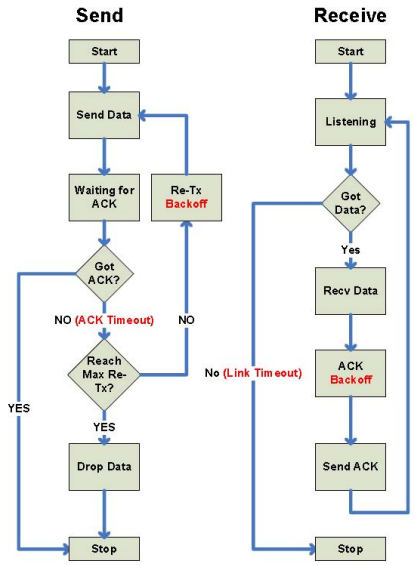


Figure 3.1: Work flow of UW-Aloha [15].

packet. Otherwise, the sender will start to prepare for new data transmission after an ACK reception.

3.1.2 Handshaking based underwater MAC

In order to reduce the collision probability in UANs, the handshaking based protocols have been proposed. They use small size of control packets to contend and reserve channel for data transmissions, in which collisions can be avoided with a proper design.

A typical handshaking MAC is Slotted FAMA [18], using RTS and CTS to reserve time slots for data transmissions. Distance-Aware Collision Avoidance Protocol (DACAP) [19] allows the sending of CTS to be fast to reduce the waiting time at the sender, hence increases the throughput of the network. Adaptive propagation-delay-tolerant collision-avoidance protocol (APCAP) [20] and Reservation-based MAC protocol (R-MAC) [21] utilize the interval time between handshaking signals to process other packets such that the channel utilization is improved. COPE-MAC [22] further improves the channel efficiency by using a parallel reservation and cyber carrier sensing scheme. Noh et al. proposed the Delay-aware Opportunistic Transmission Scheduling

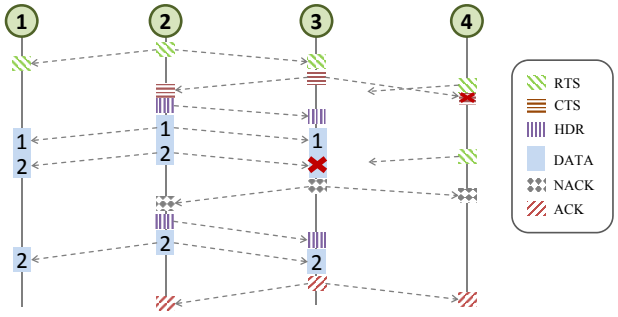


Figure 3.2: Timing of SASHA.

(DOTS) method [23] which uses topology information of the network and handshaking mechanism to improve the performance of the protocol.

Selective ARQ with slotted handshaking-based access (SASHA) [24], tested in the Atlantic Ocean experiment, is a hybrid approach of Slotted FAMA and selective ARQ, the timing of which is shown in Fig. 3.2. In order to mitigate the high overhead of handshaking process, SASHA employs packet train and selective ARQ for data transmissions. After each successful negotiation, the sender pushes multiple available packets in a train to reduce the average overhead for channel utilization improvement. If any packet gets lost, the receiver will inform the sender with a negative acknowledgment (NACK), asking the sender to retransmit the lost packets. HDR message from senders informs adjacent nodes of the following data packet retransmission for collision avoidance. This procedure will continue until an ACK is received by the sender.

3.1.3 Scheduling based underwater MAC

Scheduling based MAC protocols tend to preassign the time and/or frequency resources to nodes in a network. The classical scheduling MAC, such as TDMA, FDMA and CDMA have drawbacks of poor performance in terms of throughput and latency in networks with dynamic traffic loads.

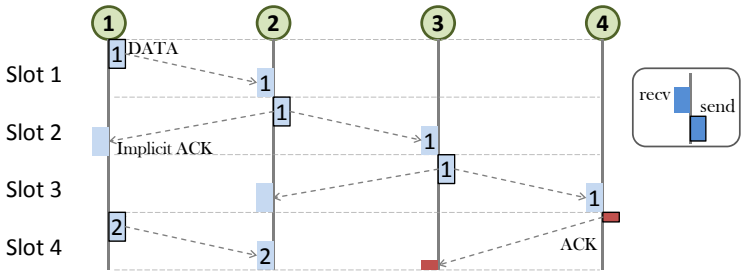


Figure 3.3: Timing of PMAC

In order to solve the spatial-temporal uncertainty problem [48] in the transmission scheduling, a couple of effective spatial or time reuse MAC protocols have been proposed. UWAN-MAC [25] leverages local synchronization to arrange the time line of each node for energy efficiency improvement, but the requirement of ultra low traffic rates in a network constraints its applications. Hybrid spatial reuse TDMA (HSR-TDMA) [26] utilizes a graph coloring algorithm to improve channel utility efficiency, but does not well address the hidden and exposed terminal problem in multi-hop UANs. Spatial-Temporal confliction graph is applied in Spatial-Temporal MAC scheduling (ST-MAC) [49] to handle the spatial-temporal uncertainty problem.

Due to the hardware limitations, the Benthos modems used in our field test do not support FDMA or CDMA schemes. The scheduling based MAC we tested in the experiments is called pipelined transmission MAC (PMAC) [47], which dedicates to string topology networks. The new feature of pipelined MAC is that it allows nodes to reuse the temporal and spatial channel resources in a multi-hop string network. As shown in Fig. 3.3, each node in PMAC is pre-assigned a time slot for data transmission. After a data transmission, the node keeps silent for the following two time slots and passes the sending opportunity to the next node. With this transmission scheme, nodes with three hops distance are scheduled to transmit simultaneously. The key points of PMAC is that any three neighboring nodes in a string network are scheduled to have

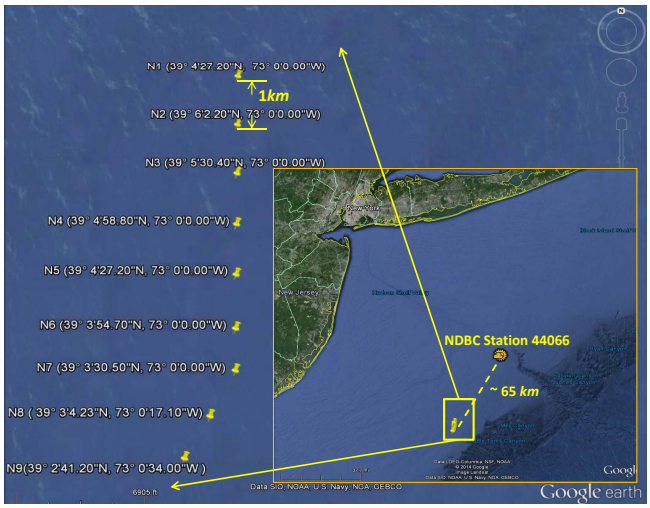


Figure 3.4: Deployment of Atlantic sea test.

staggered time slots for collision free data transmission, while maximizing the channel utilization.

3.2 Experimental evaluation on underwater MAC

In this section, I first analyze the impact of real system features on each representative MAC protocol and then compare the performance of three protocols in terms of packet behavior, node level behavior and end-to-end performances. The packet behavior analyzed in sea trials includes hop-by-hop packet delivery delays and delivery ratio. In Section 3.2.4, I define load balancing as a critical factor to evaluate node level behavior of the three MAC protocols. The end-to-end performance metrics I compare include throughput, delay and delivery ratio. Next I discuss each performance in details.

3.2.1 Experiment settings

The sea tests were conducted in the Atlantic Ocean from September 6 to September 10, 2012. In the sea tests, nine underwater acoustic nodes were deployed as a string

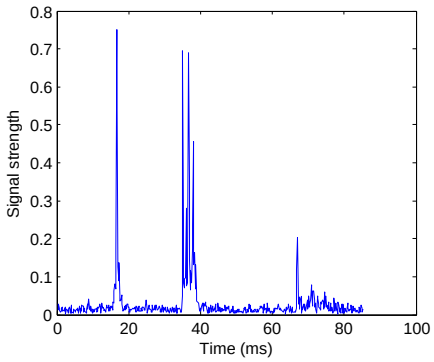


Figure 3.5: The long multipath of acoustic channel.

topology, the coordinates of which are shown in Fig. 3.4. The strip area of this experiment was about 120 km off New Jersey shore with average water depth of 80 m. The acoustic modems were deployed about 30 m below the sea surface. The closest weather monitoring center around the experiment site was NDBC Station 44066 [89], about 65 km away from the UAN nodes. According to the data collected on Station 44066, the wind speed varied significantly from 1.6 kps (knots per second) to 10.0 kps during the experiments.

In this deployment, the average distance between two neighboring nodes was about 1 km, as shown in Fig. 3.4. Due to the harsh underwater channel conditions, we had to test the modems at different spots around the target position to achieve a reliable communication at each link. Even so, we still observed significant packet losses during the experiments. The horizontal distance between two end-nodes in the network was about 7.3 km. The transmission power of modems were carefully adjusted before testing each protocol to guarantee a reliable data transmission, while constraining the communication range of each node to its adjacent neighbors to form an 8-hop network. Nevertheless, we found that the transmission range of modems during the experiments varied significantly due to the high dynamic of acoustic channel condition. Therefore, we virtually formed the 8-hop network by making each node discard packets that were sent from the nodes of more than one hop away.

Table 3.1: Traffic generation rates in different scenarios.

Protocol	4-Hop Network	5-Hop Network	8-Hop Network
`UW-Aloha`	–	–	8 bps
`SASHA`	12 bps	–	8 bps
`PMAC`	24 bps, 30 bps, 40 bps	8 bps	8 bps

Each node in the network had a Teledyne Benthos ATM-885 modem [81] connected to a surface buoy, which was equipped with a Gumstix Verdex Pro XM4 single-board computer. The Benthos modems operated at frequencies from 16 to 21 kHz with MFSK modulation scheme. In order to deal with the strong multipath effect, as shown in Fig. 3.5, the modems had to run at 600 bps or 300 bps transmission rate with 25 ms multi-path guard time. On the Gumstix micro controller, Aqua-Net framework[1] was implemented to control all three MAC protocols. Same dummy protocols on upper layers were used for all sea trials to eliminate the impact from upper layer protocols. The two end-nodes played roles as the source and the sink respectively. A Poisson traffic generator was employed at the source node, the data generation rate of which could be utilized to control the traffic load of the network. Due to the low network capacity, the traffic loads I used in experiments was low, which are between 8 and 40 bps. All three MAC protocols were tested with different network and traffic settings as listed in Table 3.1. The tested results will be discussed next.

[1] Aqua-Net framework [15] is a protocol stack with layered architecture for real UAN systems. Aqua-Net is compatible to various existing acoustic modems, including Teledyne Benthos Modems [82], WHOI Micro Modems [85] and UConn OFDM Modems [86].

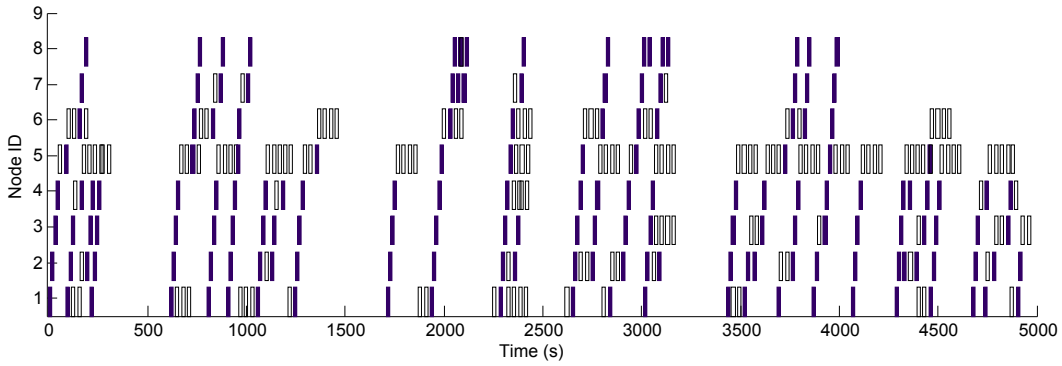


Figure 3.6: A snapshot of packet sending in UW-Aloha (Solid blue rectangular - successfully delivered to next hop; Hollow black rectangular - failed to reach next hop).

3.2.2 Direct impacts of system features

Before embarking on the performance comparison among three representative MAC protocols, I first provide some insight analysis on the impact of real system features on random access UW-Aloha, handshaking based SASHA and scheduling based PMAC.

Fig. 3.6 displays a snapshot of data packet transmissions in a one hour test for UW-Aloha. Data packets were generated on $N1$ at very low rate and were delivered to $N9$. Packets were marked as solid blue rectangular if successfully delivered to the next hop. Otherwise, failed packets were presented as hollow black rectangular. Due to the random transmission feature and low traffic rate in this UW-Aloha test, both dynamic transmission range and delayed modem transmission had minor effect on the performance of UW-Aloha. The high loss ratio and heterogeneous packet delivery, on the contrary, resulted in huge retransmissions. Especially, 8 out of 16 data packets were blocked on the link between $N5$ and $N6$ after a number of retries, owing to the worst communication reliability in the network. Besides heavy data losses, the failure on the reception of ACK packets caused unneeded retransmission along the test. The packet was retransmitted unnecessarily on $N1$ after the successful delivery at 100, 700, 2300 and 3400 second. Similar superfluous retransmissions happened almost on all nodes in the network.

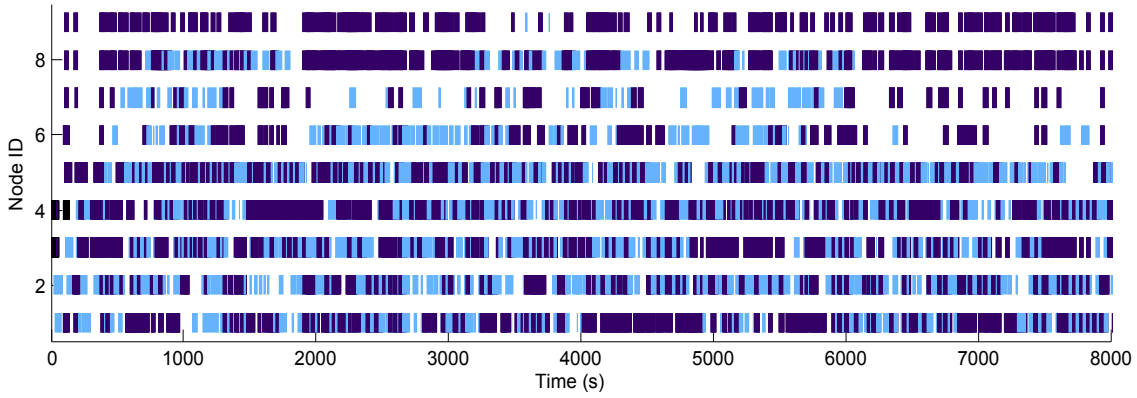


Figure 3.7: A snapshot of Backoff in SASHA (Light blue ribbon - Backoff designed in SASHA for fairness; Dark blue ribbon - Backoff when overheard RTS/CTS for collision avoidance.

In handshaking based MAC protocols, like SASHA [24], nodes overhearing RTS/CTS from other competitors will back-off in order to avoid data collisions. Packets are assumed to only be able to reach several direct neighbors within a predetermined transmission range in the conventional study. The dynamic transmission range feature of UANs introduced in Section 2.2.3 would significantly increase back-offs of SASHA in real sea networks when packets can reach nodes far away. Fig. 3.7 shows the back-off periods of SASHA in a one hour test. The data generation rate was 0.005 packet per second in this test. Light blue segments represent back-offs scheduled after one round of successful handshaking process for fairness. Dark blue segments are back-offs caused by overheard RTS/CTS to avoid collisions. Due to the transmission range uncertainty, RTS/CTS traveled to nodes of several hops away, causing heavy back-offs even at ultra low traffic rate. As illustrated in Fig. 3.7, back-offs caused by overhearing (dark blue segments) significantly dominate over back-offs after each round of data delivery (light blue segments). This is especially obvious on $N8$ and $N9$. Two nodes simultaneously transited to back-off mode when overheard packets from other distant senders very frequently during the test. These unexpected back-offs in SASHA would result in high latency in real applications.

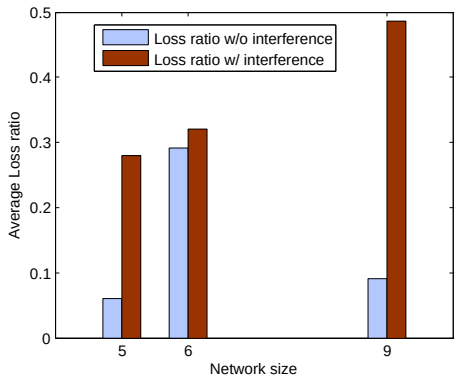


Figure 3.8: An insight of loss ratio in PMAC.

Table 3.2: Weather conditions in PMAC tests [89].

Network size	5	6	9
Wind direction[a]	258	145	208
Wind speed (kts)	9.2	3.4	6.8

[a] In degrees clockwise from true North.

Fig. 3.8 provides an insight of packet losses in PMAC. In string networks, packets were delivered in a pipelined way from the source to the destination. Nodes exact three hops away were allowed to transmit simultaneously since no collision was assumed to happen out of the single-hop transmission range. Considering the transmission range uncertainty, interference due to parallel transmissions dramatically increased the packet loss ratio in PMAC. As shown in Fig. 3.8, packet failure caused by multi-hop interference contributed the majority of the total packet losses. With the increase of network size, the chance of parallel transmissions grows, resulted in a rising loss ratio with interference. In Fig. 3.8, the loss ratio in each single test was an average of one hour experiment. The huge difference among loss ratio without interference in different tests illustrated the highly dynamic link quality in real acoustic networks. As shown in Table 3.2, both wind direction and wind speed in 6-node network test varied significantly compared to that of in 5-node or 9-node scenarios. However, whether the higher wind speed improves the communication quality of Benthos ATM-885 modems

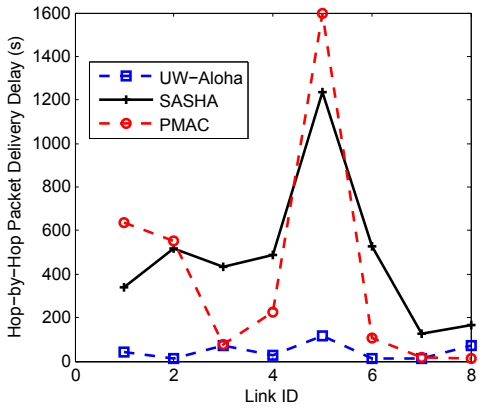


Figure 3.9: Hop-by-hop delays.

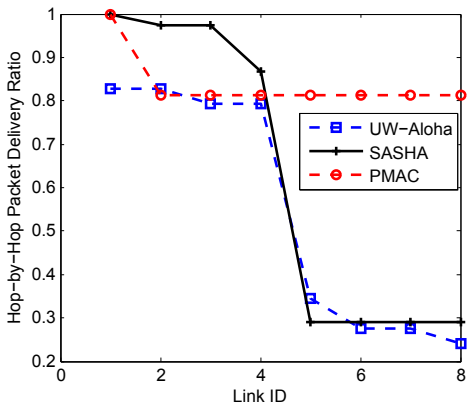


Figure 3.10: Hop-by-hop delivery ratios.

in the experiment area or what other weather conditions affected the point-to-point communications still need to be explored.

3.2.3 Packet behavior of MAC protocols

The packet behavior of the three MAC protocols studied in this subsection includes the hop-by-hop delay and delivery ratio of data packets. These two packet level performance measures serve as the foundation to analyze the overall end-to-end protocol performance and meanwhile provide insights to pinpoint the problems within the design of the protocols. The hop-by-hop delay and delivery ratio of the three protocols are presented in Fig. 3.9 and Fig. 3.10. Hop-by-hop delay is defined to be the interval between the time when the packet arrives at the sender and the time when the packet is delivered to the receiver. Hop-by-hop delivery ratio is defined to be the percentage of the packets received by the receiver at a given hop.

UW-Aloha achieved much lower delays than the handshaking SASHA and scheduling based PMAC, benefiting from the simple nature of the protocol. UW-Aloha is designed to transmit packets immediately as long as the node is in idle state. Even when it suffered huge packet losses, packets were dropped after a certain number of retries, which enabled senders to process new packets after short delays. The downside

of this packet dropping mechanism is that it leads to a low packet delivery ratio. For instance, Link 5 had a much higher packet loss ratio than other links and therefore discarded a substantial amount of packets, leading to a sharp delivery ratio drop. As discussed in Section 2.2.2, the long multipath effect was the main reason accounting for the high loss ratio in the experiment. Due to the heterogenous nature of the seabed geometry, the diversed multipath channel could result in severe packet loss on particular links. Link 5 had consistently high loss ratio than average in the tests of all three MAC protocols.

SASHA and PMAC showed comparable hop-by-hop delays, as both protocols introduce extra delays for data packet transmission. For SASHA, the two-way handshaking process is time consuming, which was 13.1 seconds in the experiment, including the long preamble, high propagation delay and guard time. If the packet loss ratio is significant, drastically long delays will be imposed on handshaking processes and retransmissions. Therefore, SASHA had larger delays than UW-Aloha and PMAC on most links. In particular, the peak delay on Link 5 was caused by the huge packet losses and retransmissions. Regarding the packet delivery ratio, on Link 5, SASHA had a sharp drop similar to UW-Aloha, but due to a different reason. Unlike UW-Aloha, SASHA retransmits until all packets are delivered. With this scheme, a large amount of packets were queued before Link 5, which had a bad link quality.

Nodes running PMAC take turns to send packets. Each cycle took about 32.9 seconds in our experiment. If retransmission is needed, it has to wait 32.9 seconds for a new transmission cycle regardless whether its neighbor nodes have sending task in the assigned slots. The significant amount of packet losses and retransmissions on Link 5 caused a remarkable peak in packet delivery delay. This is a limitation of statistic scheduling design. However, since the whole running time was used for data packet

transmissions in PMAC, nodes had more transmission opportunities than nodes in SASHA, and therefore achieved appealing delivery ratios on all hops along the path.

3.2.4 Node behavior of MAC protocols

In the node behavior analysis, I compare the data sending and receiving behaviors of each node among the three MAC protocols. The number of packets sent and received in a two and half hours test is listed in Table 3.3. I define load balancing factor in Equation (3.1).

$$F_B = \frac{(\sum_{i=1}^{n} N_i)^2}{n \sum_{i=1}^{n} N_i^2}, \tag{3.1}$$

where n is the number of nodes and N_i is the number of packets processed at node i. It is a different metrics from fairness [90], which is defined as an equal share of bottleneck. Load balancing refers to the balanced sending or receiving actions among nodes in the multi-hop network. Load balancing is a preferred feature in a network when all nodes have same traffic load, which was true in the experiment.

Table 3.3 lists the balancing factor of three MAC protocols. Transmission balancing plays an important role in determining the network lifetime. Balanced transmission can help to avoid the early depletion of a node because of unbalanced heavy load, thus enhance network connectivity. Reception balancing, on the other hand, can help to avoid over-crowded region in the network. This feature is crucial for MAC protocol performance since the collision probability relies on the traffic rates. Severe interference would happen in the over-crowded region.

UW-Aloha had poor balance on both sending and receiving events, since nodes closer to the source had more packets to receive and transmit than nodes closer to the sink, especially when a large amount of packets were dropped on Link 5. SASHA had better transmission balancing benefiting from the handshaking mechanism. Data packets could be sent out only when reservation is successful, which reduces unnecessary

Table 3.3: Number of packets sent and received along the path.

Sender ID		Number of Data Packets									F_B
		N1	N2	N3	N4	N5	N6	N7	N8	N9	(%)
UW-Aloha	Send	96	90	81	73	163	77	37	35	\	82.6
	Recv	59	49	49	47	47	20	16	16	14	80.9
SASHA	Send	66	75	56	47	46	49	30	29	\	90.0
	Recv	59	52	51	51	45	15	15	15	15	79.5
PMAC	Send	175	239	98	181	334	132	67	49	\	76.4
	Recv	59	59	48	48	48	48	48	48	48	99.2

data transmissions. However, receiving balancing factor is still low due to the severe packet losses. Similar to UW-Aloha, much fewer packets traveled through Link 5, due to the time consuming two-way handshaking and retransmissions.

For PMAC, on the contrary, all nodes in the network have equal slots to transmit. Unbalanced sending was caused by retransmissions. Since PMAC is designed to be collision free, more significant packet losses and retransmissions than UW-Aloha and SASHA in the test was possibly caused by the transmission range dynamics. PMAC is based on the assumption that nodes two hops away are unable to reach each other and therefore allowed to send simultaneously, which should be the truth in the field. However, due to the communication range uncertainty, packets can reach nodes further than two hops, leading to unexpected collisions. Another serious problem for scheduling based MAC is the delayed modem transmission. When a packet is failed to be pushed out at scheduled time due to the busy terminal problem, unexpected collisions also occur. The receiving process, on the other hand, achieved high balancing rate in PMAC. Nodes almost received similar number of packets along the path and led to a higher end-to-end throughput than the other two protocols, which will be discussed in Section 3.2.5.

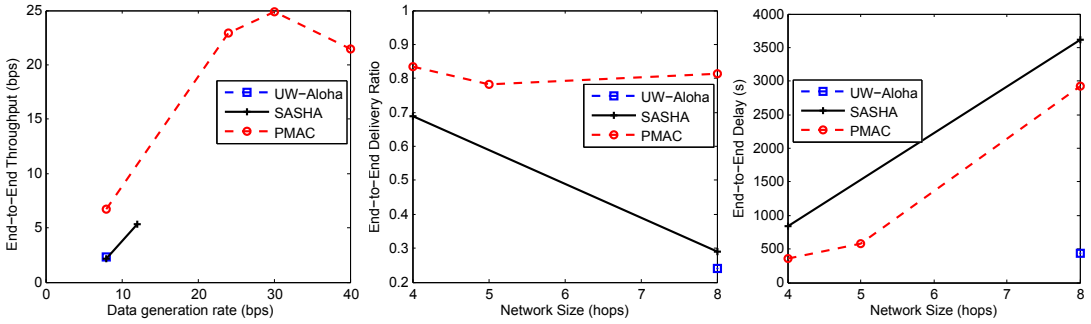
3.2.5 End-to-end performance

The end-to-end performance metrics I focus on in the comparison are throughput, packet delivery ratio, and delays.

3.2.5.1 End-to-end throughput

End-to-end throughput is the most direct metric to evaluate the network performance. As shown in Fig. 3.11(a), UW-Aloha got the lowest throughput even at ultra low traffic rate. Because of time limitation, we did not get chance to conduct further test for UW-Aloha. The low throughput was a result of the packet drop mechanism. As revealed in Table 3.3, a large number of packets failed to reach $N6$, and thereby leaded to low end-to-end throughput and delivery ratio performance for UW-Aloha. SASHA achieved similar throughput performance with UW-Aloha at the low traffic rate. SASHA handles data packet interference better than UW-Aloha using RTS/CTS reservation, but with a penalty incurred by high handshaking delays. The throughput of SASHA increased when the network had higher traffic rates. However, SASHA significantly underperformed PMAC. Since the whole time in PMAC was assigned for data transmission, the throughput performance linearly grew with the increasing network load before the PMAC saturated.

The highest throughput of PMAC only depends on the modem transmission rate and channel quality. Due to the pipelined scheme, the hop with worst channel performs as the bottleneck. As the traffic rate grows, the end-to-end throughput of PMAC saturated at about 25 bps, which indicates the maximum throughput of the network under the same experiment setting and channel conditions. We did not get chance to test the traffic rate beyond 12 bps for SASHA due to the time limitation. However, given the significant overhead of SASHA on handshaking process, the 12 bps traffic rate used in the experiment had been very close to the capacity, if not reached yet.



(a) (b) (c)

Figure 3.11: End-to-end performance comparison. (a) Throughput, (b) Delivery ratio, (c) Delay.

3.2.5.2 End-to-end delivery ratio

End-to-end delivery ratio is an important metric related to the network reliability. Due to the high cost of transport layer retransmission in UANs, link level reliability becomes an essential feature for MAC protocols. In Fig. 3.11(b), I compare three MAC protocols in terms of delivery ratio performance. End-to-end packet delivery ratio relies on the network size. With a larger number of hops along the path, packets are more vulnerable to losses.

UW-Aloha had as much as 75% packets loss along the path after a limited number of retransmissions. The huge packet losses were caused by the extensive collisions and bad channel condition. When the network was as small as 4-hops, SASHA successfully delivered 70% of packets generated. However, when the network size increased to 8-hops, the delivery ratio drastically reduced to 28%. As Link 5 was very unstable in the 8-hop network, a quite large number of packets were stuck in the middle of network. Even though SASHA is designed to be able to avoid data packet interference, the time consuming handshaking process and high packet losses lead to low capacity on data packet delivery, which is the main reason for the low delivery ratio in the 8-hop network. PMAC, on the other hand, achieved the highest end-to-end reliability. Benefiting from

the scheduling mechanism, nodes had time to process more packets since no contention or reservation delays were introduced in PMAC.

3.2.5.3 End-to-end delay

End-to-end delay grows with the increase of network size, since longer time would be required to reach the destination if the sink is further away. Fig. 3.11(c) shows the delay performance of the three MAC protocols.

According to the design of UW-Aloha, packets are pushed out with minimum delays. Even though superfluous packets were lost in the middle of network, the delivered packets were able to reach the destination with small end-to-end delays, as shown in Fig 3.11(c). SASHA and PMAC had much more significant end-to-end delays than UW-Aloha. This conclusion is consistent with the results of hop-by-hop delays in Fig. 3.9. Since both protocols are designed to deliver all packets with unlimited retries. When one link (Link 5) was bad, the substantial retransmissions led to dramatic delays. Delays of both SASHA and PMAC linearly increase with the network size according to the test results. Similar to the throughput performance, SASHA was inferior to PMAC on delay performance. This is also caused by the time consuming handshaking process considering the low sound speed and long preamble in acoustic modems.

3.2.6 Discussions

In previous subsections, I have analyzed how the real system features affect the performance of three representative underwater MAC in sea experiments. In this subsection, I discuss how these real system features affect general MAC protocols and provide some suggestions on how to address the observed issues in practical underwater MAC design.

The long preamble feature challenges almost all protocols including MAC, routing, synchronization and localization, especially for those with small control packets. For the purpose of synchronization, signal detection and AGC control, a preamble is normally used to lead a physical layer frame. This implies that "short" control packets are not short anymore and will equally suffer heavy collision as data packet in the network. Moreover, the long preamble will remarkably increase the energy consumption on control transmissions. An efficient usage of control messages will be a necessary for practical protocol design in UANs.

The high packet loss ratio brings grand challenges to ubiquitous underwater MAC protocols. Even though most MAC protocols employ retransmission mechanism to deal with packet losses, it becomes inefficient if we consider the long preamble length, high propagation delays and high energy consumption with retransmissions in real systems. In this case, network coding technique [91] becomes a promising technique benefiting from the broadcast nature of UANs. The communication reliability could be improved to some extent with the error recovery mechanism in network coding. However, the time-varying feature of UANs requires dynamic coding rate, which is still an open issue.

Similarly, the heterogeneous packet delivery has negative effect on general MAC protocols. For example, in the ARQ based MAC protocols, both the data and acknowledgment loss will lead to data packet retransmissions. This implies that a worse feedback channel (severer acknowledgment loss) could extensively introduce unnecessary data retransmissions and result in considerable waste on energy and channel resources. To counter the adverse outcome of heterogeneous channel condition, one viable solution is to employ independent coding rates for channels with different link qualities. However, this introduce further complexity for MAC protocol design.

The spatial and temporal communication range uncertainty changes the communication and interference area in real experiments, which is generally assumed to be fixed

and homogenous in theoretical and simulation studies. For scheduling based MAC protocols, the communication range uncertainty can incur undesired interference to nodes that are multi-hop away if they fail to consider the dynamic neighborhood issue. For other MAC protocols, the dynamic interference area leads to a gap between simulation results and the real performance in sea experiments. The communication/interference range uncertainty issue can be mitigated by dynamic scheduling, such as adaptive power control and channel hopping. However, the online adaptation is difficult to implement in real underwater networks, where the negotiation throughout the network is slow and inefficient even if a common control center exists.

Multi-hop interference is very hard to eliminate considering the spatially and temporally communication range uncertainty. Due to the low spreading loss of acoustic communications compared with radio communications in terrestrial networks, strong interference from distant senders would significantly reduce the communication reliability in UANs even with good channel quality. In addition, the strength of interference across the network is heterogenous due to the unbalanced transmissions among different senders, which makes the interference harder to deal with. It becomes an open issue to effectively address distant interference either with more advanced communication schemes or more effective transmission scheduling in MAC protocols.

The delayed data transmission on acoustic modems are mainly resulted from two factors, namely the clock drift and busy terminal problem. The clock drift of microcontroller depends on the remaining power of battery, the surrounding temperature, humidity and other environmental factors, which vary a lot in the sea surface. Thus real underwater networks call for time synchronization with low overhead, especially for long-term applications. The impulsive delays caused by busy terminal problem can make the collision avoidance futile, especially in the network where nodes experience heavy overhearing. Adding large guard time could reduce the chance of collisions when

the data transmission is undesirably postponed due to the busy terminal problem, however, with a penalty of increased network latency. To avoid scheduling transmissions when the modem is busy, the MAC protocols require to have knowledge of the modem status in real time. But so far the timely response to the upper layer is not well provided with existing acoustic modems.

3.3 Practical underwater MAC design

Through experimental study in Section 3.2, I have discussed the advantages and limitations of different MAC protocols. More specifically, random access based MAC has the shortest delivery delay but the lowest throughput and worst reliability. Therefore, MAC protocols based on random access are suggested to the delay-sensitive network with ultra-light traffics. Scheduling based MAC has the highest throughput. However, it is strongly limited to network topology and has the worst adaptivity to different applications. Compared to random access and scheduling based MAC, Handshake based MAC has moderate throughput, delivery ratio and high delays and is suitable to general delay-tolerant applications. Therefore, I stick to handshake based MAC protocols and tackle to address the challenges from practical issues. The content in this section is partially based on my previous work published in [88][3] .

Most of existing handshake based MAC protocols [20,92], including SASHA, are all sender initiated handshaking approaches, implying that the sender starts negotiation by sending out a RTS message for channel reservation. Another category of MAC protocols reverses the handshaking process, and is called receiver-initiated (RI) scheme [93–95]. For instance, Nitthita Chirdchoo et.al proposed receiver initiated packet train (RIPT) protocol [94] for UANs. In RIPT protocol, the handshaking process is initiated by the receiver which sends a request-to-receive (RTR) message to poll data from neighbor

[3] © 2013 ACM. Reprinted with permission from Lina Pu, Traffic estimation based receiver initiated MAC for underwater acoustic networks, ACM WUWNet, Nov. 2013, doi: 10.1145/2532378.2532412

nodes. Compared with sender initiated MAC, receiver initial protocols have following advantages:

1) *Significantly reducing the overhead on control messages.* In the practical implementation, control packets (RTS, CTS and ACK) are usually used in MAC protocols to avoid collisions and to guarantee a reliability data transmission. Due to the long preamble in acoustic modems [43, 96], the impact of overhead traffic generated by the control packet transmission is much severer in UANs than in radio networks. As shown in following examples, where assume the length of preamble, the size of useful data and data transmission rate of modem are 0.5 s, 200 Bytes and 3 kbps, respectively, the control packet overhead in the single-send-single-receiver case is as high as 167%. By adopting parallel reservations in the single-send-multi-receiver case, where a single sender reserves data communications with multiple receivers, the average overhead can be reduced to 123%. By contrast, in receiver initiated approaches, the overhead of control packets is only 78%. The improvement comes from the fact that the receiver needs only one ACK to inform all senders of the successful data reception in receiver-initiated MAC approaches.

- single-sender to single-receiver

  $(RTS + CTS + ACK)/Data = 0.56 + 0.56 + 0.56 \approx 167\%$

- single-sender to multiple-receiver

  $(RTS + 5CTS + 5ACK)/5Data \approx 123\%$

- multiple-sender to single-receiver

  $(RTR + 5ATS + ACK)/5Data \approx 78\%$

2) *Enabling data aggregation at the link layer.* In receiver initiated MAC protocols, since a receiver requests and collects data packets from surrounding neighbors in one round communication, data packets can be merged at the link layer and therefore significantly reduces the traffic load of the whole network. In addition, the link layer

data aggregation is also beneficial to the applications with data fusion [97]. As an assistance to the end-to-end data fusion, the local information gathering is essential to support a fast response to the dynamic of networks.

Despite the aforementioned advantages it has over traditional handshaking methods, receiver initiated MAC protocols still face challenges in the implementation. Data polling, defined as the process the receiving node uses to retrieve data from sending nodes, is one of the major problems. Since the receiving node is unaware of the status of surrounding senders, how to design an efficient and timely data polling scheme becomes a big challenge. There are two fundamental questions a data polling scheme has to answer: (1) "When will the data packets be available for a receiver to poll?" and (2) "How many data packets should a receiver to request?"

Furthermore, the data polling scheme has to be adaptive to both the dynamic traffic pattern and various application requirements in UANs. On the one hand, the number of queued packets at the link layer is a random variable that may vary with time and among different senders, due to the combined effects from all the upper layers [98], including network layer, transport layer and application layer. On the other hand, due to the limited computation and memory capacity of UAN nodes, it is impractical to approximate the traffic distribution from a large amount of data samples.

In this section, to address the problem of adaptive data polling I propose a receiver initiated MAC protocol, called traffic estimation based receiver initiated MAC (FERI MAC) [88] for UANs. In FERI MAC, I employ the cumulative density function (CDF) inversion sampling technique to pull a number of samples from the large historical dataset and utilize the most recent observed data to catch the trend of the network traffic. In this way, an accurate estimation of traffic pattern can be achieved to support the traffic adaptive data polling mechanism in FERI MAC. FERI MAC can achieve a good energy efficiency and channel utilization by delaying the data request

at the receiver and adjusting the channel resource assignment among different senders. Therefore, unlike prior RI MAC protocols which are limited to certain traffic patterns, FERI MAC can be applied to networks with arbitrary traffic rates while maintaining a high energy efficiency and channel utilization per user request.

3.3.1 FERI MAC design

FERI MAC employs a receiver initiated handshaking procedure composed of four phases. The procedure starts with the receiver sending out a RTR message when it intends to ask for data from its immediate neighbors, which is Phase 1 in Fig. 3.12. RTR message consists of the current receiving node address A_{recv}, the next-hop address A_{next}, the polled sender addresses A_{send} and the time slots assigned to each sender N_{slot} for the following data transmission phase. Node with address A_{next} is the next-hop destination of the current active receiver. In multi-hop networks, this information performs as data sending request of node A_{recv} to inform node A_{next} to start polling timely after current round of communication. In this way, the packets can be delivered to the destination smoothly with shorter queuing delay. The slot assignment N_{slot} is associated with traffic estimation for each sender, which will be introduced in Section 3.3.3.

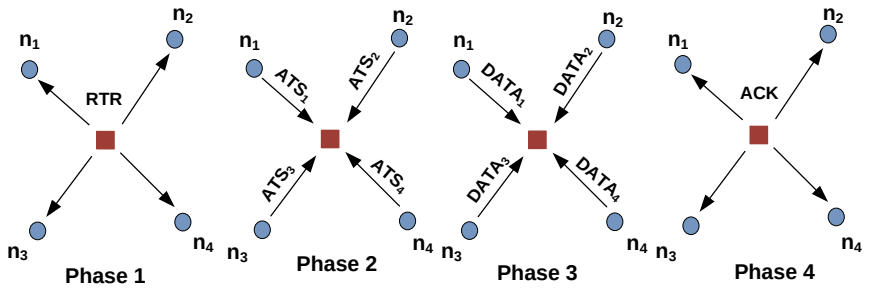


Figure 3.12: Four phases in FERI MAC.

In Phase 2, the invited senders need to respond with an available-to-send (ATS) message to establish a data transmission session. With parallel reservation, the transmission times of ATSs from multiple senders should be staggered and follow the order as scheduled in the RTR packet. The ATS message includes the sender address A_{send} and the number of packets N_{pkt} that the sender has actually queued. This feedback, as an input to the traffic estimation algorithm discussed in Section 3.3.3, helps the receiver with the network traffic estimation.

In FERI MAC, the collisions can be avoided by RTR and ATS messages exchanging. During the handshake, the one-hop neighbors of both the receiver and senders are notified of the ongoing data transmission and therefore will act properly to avoid collision to the ongoing communication. Actually, compared with sender initiate based approaches, FERI MAC is more effective in protecting the receiver from packet collisions. This is because all the potentially interfering nodes of the receiver are exposed and informed at the very beginning of the handshaking process. By contrast in traditional sender initiated MAC when the sender starts the handshake process, some potentially interfering nodes might be two hops away, known as hidden terminals to the sender, which makes the data reception more vulnerable to the interference from hidden nodes.

In the data transmission phase (Phase 3 in Fig. 3.12), time is divided into mini slots, which are assigned to the senders in Phase 1 by the receiver. The senders will send data packets in the scheduled time slots if it is able to transmit. Since the slot allocation is based on the traffic estimation, which may not be accurate, some of the allocated slots might be wasted if the sender does not have enough packets to send out; or remaining packets might be queued at the sender if insufficient slots are scheduled. For such reason, FERI MAC is limited to the applications without strict delay requirements. However, as long as we can get a good approximation of the traffic distribution of

sending nodes, which is stationary at least in the short term, FERI MAC can achieve a desirable energy efficiency and channel utilization based on the adaptive data polling which will be discussed in Section 3.3.2.

In the last phase (Phase 4) of FERI MAC, the receiver replies an integrated ACK message to inform the successful data reception, instead of multiple ACKs from each receiver in sender initiated protocols. Considering the long preamble in acoustic modems, the reduced number of ACK packets significantly decreases the power consumption and thus extends the network lifetime.

FERI MAC conserves energy in three ways. First, the receiver initiated reservation is more effective in preventing the data packet collisions since all the potentially interfering nodes of the receiver are informed at the very beginning of the handshaking process. In addition, FERI MAC employs parallel reservation and packet train to reduce the overhead of handshaking control messages. Finally, the number of ACK packets is reduced significantly compared to sender initiated protocols.

3.3.2 Adaptive data polling

Data polling mechanism aims to address two fundamental issues in receiver initiated MAC protocols: when to request data from senders and how much data to request. It becomes a big challenge since the receiver usually lacks information of senders. Theoretically we can allow a sender to inform the receiver its current status by sending some update packets. However, considering the long preamble in acoustic modems, such a strategy would incur nontrival overhead.

To address these two issues, I proposes an adaptive data polling scheme with the assistance of the link layer traffic estimation in FERI MAC. The receiver estimates the traffic distribution of a sender based on the historic traffic information, which is obtained from the past ATS packets. With the help of traffic estimation, an appropriate

data polling frequency can be determined to achieve a desirable delivery delay and energy efficiency. Traffic estimation is also beneficial for deciding the number of packets to poll from the senders.

3.3.2.1 When to poll data

There is a trade-off between the energy efficiency and delay performance with respect to the data polling frequency. In UANs with power constrain, high energy efficiency is more preferred to extend the network lifetime. In order to achieve a controllable performance, I set up a threshold of control packet overhead E_{th}, which is defined as the total power consumption on control packets over that of on the data transmission during one round handshake, and a threshold of one-hop queuing delay D_{th}, which is defined as the delay for a packet awaiting for the transmission.

In FERI MAC, a node will start the data request in three cases. 1) If a node is the next-hop destination of the current active receiver, the node will start the data polling as soon as it can. Thus the packets can be forwarded to the final destination smoothly in multi-hop communications. Note that as described in Section 3.3.1, the RTR message, which includs the A_{next} information will notify the succussive node of the coming reception. 2) A node will initiate the handshake if the expected energy efficiency E_{xpt} reaches the defined threshold E_{th}. The receiver node estimates E_{xpt} based on the traffic distribution of each sending node. In this way, a baseline energy efficiency E_{th} can be achieved in FERI MAC. 3) A node will request data from neighbors when the time passed since the last communication exceeds the delay threshold D_{th}. In a network with a low traffic rate, the time it would take to accumulate enough packets for the packet train can be too long to accept. To avoid this situation, I add D_{th} to guarantee a maximum delay of D_{th} in one-hop communications.

Algorithm 1 When to poll Data

```
if A_index == A_next then
    Poll Data when the current handshaking ends;
else
    while E_xpt ≥ E_th || max(D_i) ≤ D_th do
        Waiting time ++;
    end while
    Poll Data if channel is idle;
end if
```

3.3.2.2 How much data to poll

In UANs, the traffic in different senders may vary significantly since it includes both the self generated data and the packets forwarded for other nodes, which is determined by the network topology and the routing protocol. Also the traffic at the same sender may change dramatically with time because of the dynamic in both the network data generation and the routing algorithm.

Since the amount of packets of each sender is a random variable varying with time, the receiver can hardly know how many packets to invite from senders without any extra communication. However, with a traffic distribution knowledge, the receiver will be able to assign time slots for each sender to guarantee a predetermined delivery percentage, P_{del}, in each round of communication. A trade-off between the channel utilization and delivery delay can be achieved by adjusting P_{del}.

- The Delivery Percentage, P_{del}: When the amount of packets in the polled node follows a given distribution $F(x)$, there is a probability P_{del} that all packets can be covered with the assigned slots, L.

$$P_{del} = \int_0^L F(x)dx$$

By adjusting the percentage P_{del}, we can achieve a trade-off between channel resource utilization and delivery delay. A high P_{del} reduces the packet residual probability and thus decreases the average delivery delay. However, it may increase the probability

of channel resource wasting because of the potentially over assigned time slots. A low P_{del}, on the contrary, would lead to a long average delivery delay while maintaining high channel utilization. Choosing an appropriate P_{del} is therefore an optimization problem for the receiver initiated MAC protocols, which will be introduced later.

3.3.3 Traffic estimation for FERI MAC

In this subsection, we I present that how to estimate the link layer traffic with statistical method.

During the handshaking process, the average packet number within the time interval between two successive data requests is informed to the receiver, serving as an input for the traffic estimation. A standard method for an approximate distribution is the empirical distribution of the observed data from all the history records, e.g., $x_1, x_2, ..., x_N$. We can get a better estimation with a larger sample size N. However, because of the traffic dynamics in the time domain and the constrained computation and memory capacity in underwater nodes, we can only expect a relatively sparse dataset for a rough traffic model approximation.

On the one hand, a larger sample size leads to a more accurate traffic estimation and a more efficient channel allocation for the FERI MAC. On the other hand, a larger dataset slows down the adaptation of traffic model to the varying link layer traffics and aggravates the computation overhead of underwater nodes. Instead of keeping all the history samples, $x_1, x_2, ..., x_N$, I utilize the CDF inverse sampling technique to poll samples $\hat{x}_i, i = 1, ..., K$ from the traffic distribution $F(x)$ as a representative to the large dataset.

- *CDF Inverse Sampling* [99]: Let $G(x)$ be the CDF of random variable x, which has the PDF $F(x)$. If random variable y comes from uniform distribution $U(0, 1)$, then

the random variable $\hat{x} = G^{-1}(y)$ follows the same distribution with x. $\hat{x}_1, ..., \hat{x}_N$ become the samples from the distribution $F(x)$.

CDF inverse sampling is a simple but effective method when the distribution is univariate and the inverse CDF is easy to get. In many cases when inverse CDF can not be solved analytically, we can approximate the actual CDF with a piecewise linear function based on the sample set from the distribution.

In order to catch the trend of the traffic when it varies, I choose the most recent M records $x_{N-M+1}, ..., x_N$ to approximate the traffic distribution together with samples $\{\hat{x}_i\}$. The sampling window of size M represents the most recent trend of the network traffic, while the resampled data $\hat{x}_1, ..., \hat{x}_K$, out of the preceding records represents the historical information of the traffic distribution. With the small dataset $\hat{x}_1, ..., \hat{x}_K, x_{N-M+1}, ..., x_N$, we can then estimate the traffic distribution with an affordable computation overhead, e.g., kernel density estimation method.

Algorithm 2 Traffic Estimation in FERI MAC

```
while new record x_{N+1} arrives do
    if S_WIN ≥ M then
        Move out x_{N-M+1}, ..., x_N in the sampling window;
        Resample from x̂_1, ..., x̂_K, x_{N-M+1}, ..., x_N
    end if
    Add x_{N+1} in the sampling window;
    Build PDF with x̂_1, ..., x̂_K and records in sampling window;
end while
```

When the new record x_{N+1} arrives, the sampling window moves forward and leaves the sample x_{N-M+1} out of the window. I resample the remaining records $\hat{x}_1, ..., \hat{x}_K$ and x_{N-M+1} to keep a non-increasing dataset. To reduce the frequency of resampling process in the traffic estimation, a simplified procedure is illustrated in Algorithm 2. I employ a sampling window with a changing size. The size of the sampling window S_{WIN} increases with the coming records. All the past records in the sampling window are

moved out when $S_{WIN} \geq M$ and resampling with the historical $\hat{x}_1, ..., \hat{x}_K$ is performed to get a new representation for the historical distribution.

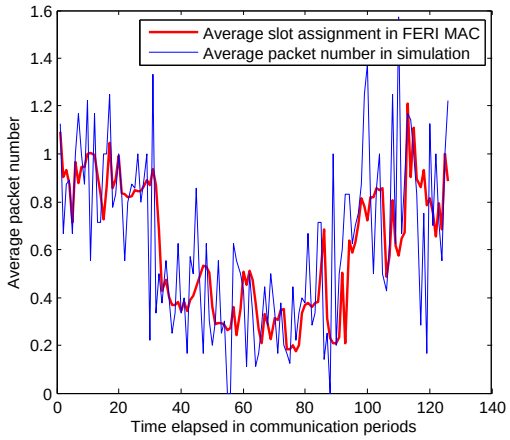


Figure 3.13: Adaptive slot assignment in FERI MAC with varying traffic rate.

Depending on the traffic estimation as shown in Algorithm 2, the receiving node estimates how much data to request from each sender and assigns time slots for their data transmissions. Fig. 3.13 demonstrates the effectiveness of the traffic estimation scheme in FERI MAC, where I set the sampling window size M to be 10 and the resampling window size K to be 5. In this example, the number of the assigned slots to a single sender by the receiver as well as the actual number of packets held by that very sender are shown in Fig. 3.13. We can see that with the help of the traffic estimation, these two achieve a good match. An overall 88% channel utilization is achieved in this example when trying to assign the slots to guarantee a 50% probability of packet coverage in the transmission.

3.3.4 Performance evaluation

In this subsection, I use the energy efficiency, channel utilization and one-hop delivery delay to evaluate the performance of FERI MAC.

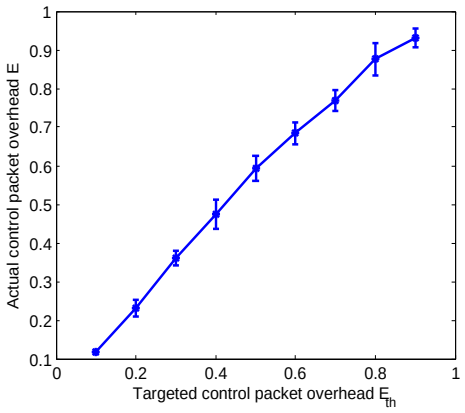


Figure 3.14: Achieved energy efficiency with respect to targeted E_{th}.

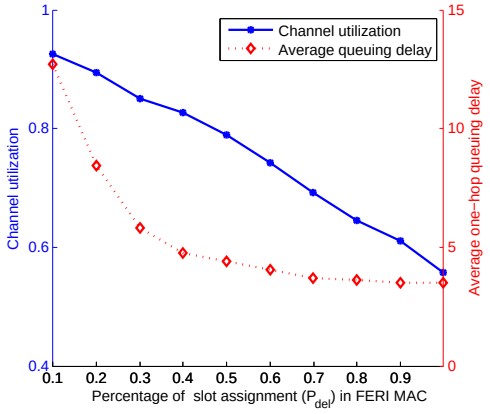


Figure 3.15: Trade-off between channel utilization and one-hop delivery delay.

The energy consumptions of the control and data packets in my simulations are calculated based on the UConn OFDM Modem in [43]. The energy efficiency performance of FERI MAC is controlled by adjusting the targeted control packet overhead E_{th} in Algorithm 1. Being aware of the traffic distributions of all surrounding senders, the receiving node that initiated the handshaking process is able to adjust the frequency of data polling to reach the required energy efficiency. If the traffic rate is relatively low, the waiting time is increased to allow more packets to be accumulated at the senders. Otherwise, the receiver will request the data more frequently in order to decrease the delivery delays. Fig. 3.14 shows a good fit between the targeted and the actually achieved control packet overhead in simulations. The good consistency between the desired E_{th} and the achieved E verifies the effectiveness of both the energy efficiency control and the traffic estimation in FERI MAC.

The transmission slot allocation among senders based on the traffic estimation is a trade-off between channel utilization and delivery delay performance, which is achieved by adjusting the delivery percentage, P_{del}. The channel utilization linearly decreases with the increase of P_{del}, as shown in Fig. 3.15, coming up with a reduced delay in an inverse proportional way. Notice that when a small number of slots are assigned

to senders, a considerable extra queuing delay is introduced in the communication. This queuing delay decreases dramatically with the increase of P_{del} in the beginning. However, it does not have significant reduction when P_{del} is beyond 0.5. This queuing delay is caused by the design of the receiver initiated scheme such that the senders wait for data polling from the receiver. Considering the high energy efficient feature of FERI MAC and the relatively long queuing delay as a penalty, FERI MAC is suggested to be applied to energy constrained UANs with delay tolerant applications.

In Fig. 3.16 and Fig. 3.17, I compare the energy efficiency and delay performance of FERI MAC with RIPT, an underwater receiver initiated MAC with packet train. RIPT utilizes a four-way handshaking letting the invited senders to inform the receiver of the number of packets with an extra control packet. Compared with RIPT, FERI MAC presents a significant advantage on energy efficiency as revealed in Fig. 3.16, at the cost of a longer one-hop delivery delay, as shown in Fig. 3.17. In the FERI MAC simulation with $E_{th} = 0.2$, a desired energy efficiency performance is achieved under a wide range of network traffic loads, which is much lower than that of RIPT MAC, especially at low traffic rates. However, a longer one-hop delivery delay is introduced in the FERI MAC, due to the less frequent data polling when the network has a light traffic load.

The performance of RIPT heavily relies on the network traffic load. When the traffic rate is low, there are only a limited number of data packets available in the senders for each handshaking communication. This over-frequent data polling in RIPT MAC results in a relative high control packet overhead and thus a poor energy efficiency. The efficiency performance improves at high traffic loads, but is still worse than FERI MAC. When we add the traffic estimation to RIPT MAC, the combined protocol achieves a significantly improved energy efficiency over RIPT as shown in Fig. 3.16. With an estimation of the number of packets at each senders, the receiver notices the low traffic

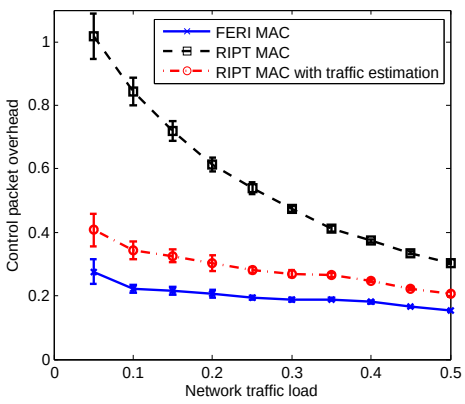


Figure 3.16: Overhead of control packet with respect to network traffic load.

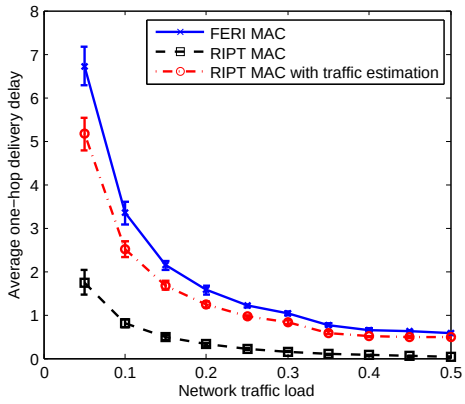


Figure 3.17: Hop-by-hop delivery delay with respect to network traffic load.

rate of the network and tends to slow down the handshaking process to allow packet accumulation at the senders. Even though a longer delay is resulted in by the adaptive data polling scheme, the substantial energy efficiency improvement is more promising to the power constrained UANs. This efficiency advantage over RIPT also verifies the effectiveness of the traffic estimation assisted adaptive data polling scheme for the receiver initiated MAC protocols.

3.4 Summary

In this chapter, I evaluated three representative MAC protocols, namely, random access based UW-Aloha, handshaking based SASHA and scheduling based PMAC, in sea trails. Through experiment results the direct effects of real system features of UANs on the MAC protocol performance were explored. I also analyzed and compared the MAC protocols on end-to-end throughput, end-to-end delay and end-to-end delivery ratio in different network settings. Based on the field test results, I studied the advantages, shortcomings and limitations of three MAC protocols and how they work in real systems. Following this, I discussed the impact of UAN features on general underwater

MAC protocols design in hope of providing some meaningful insights into practical MAC design for real multi-hop networks.

To tackle the aforementioned challenges from real systems, I proposed a practical MAC, called traffic estimation based receiver initiated MAC (FERI MAC), for underwater acoustic networks. The adaptive data polling mechanism implemented in FERI MAC addressed two fundamental issues in receiver initiated MAC: when to poll data from senders and how much data to request. Especially FERI MAC can achieve an user-desired energy efficiency by adjusting the data polling frequency. Also it achieves a trade-off between the channel utilization and packet delivery delay with the adjustment of the amount of packets to poll. Further, the adaptive data polling scheme makes FERI MAC applicable to networks with arbitrary traffic patterns. Simulation results demonstrate the effectiveness of FERI MAC in terms of achieving the desired energy efficiency as well as balancing the channel utilization and delivery delay. Also simulation results show the significant advantage of FERI MAC on energy efficiency over conventional receiver initiated MAC without adaptive data polling.

Chapter 4

Impact on Underwater Synchronization

Due to the application requirements and the severe resource constraints in an underwater environment, most of UANs are designed as distributed system. Nodes in these networks usually require a time synchronization service for sensor data collection [50], network localization [51] and coordination in MAC and cooperative communications [52].

Time synchronization has been a research area of long history [54]. Many synchronization protocols have been proposed and tested in the TWN [55–59]. The source of delays and uncertainties in message delivery have been extensively studied in the radio system. The effect of these uncertainties on synchronization protocols have been evaluated and schemes to improve precision have been proposed [56, 57]. However, the environments of the oceans and grounds are very different, leading to distinct designs between acoustic and radio systems. Due to the unique features of UANs, the territorial time synchronization protocols may need an overhaul before using them efficiently in underwater environments.

Recently, the time synchronization for UANs has drawn people's attention along with the development of underwater communication and networking technology. Synchronization protocols, such as [31, 60, 61], have been proposed for the high latency

and mobile underwater networks. In these works, the long propagation delay of the acoustic signal and the mobility of the acoustic nodes are carefully studied, and the errors caused by these unique features are well compensated.

Due to the high cost of energy and time on the real sea experiments, a majority of current synchronization protocols designed for UANs still stay on theoretical analysis and simulation evaluations. Their actual performance in real acoustic systems is rarely known, and few attempts have been made in the literature to evaluate them with experiments. Meanwhile, the radio model on the source of errors in synchronization is often used in existing UAN synchronization design. However, the differences between acoustic and radio communication systems may preclude the direct use of the radio model in underwater networks. The decomposition of message delivery delays in real acoustic system and their effect on synchronization protocols need to be measured and tested. Along this direction, I conduct experiments and evaluate representative synchronization protocols in real acoustic systems.

4.1 Background and related work

Time synchronization has been a critical piece of infrastructure in distributed systems for a long history. Numerous protocols have been proposed and used in a wide spectrum of applications, including networking, localization, target tracking, environment monitoring and cooperative communications.

In Internet, network time protocol (NTP) [54] has been widely used to synchronize the client computers with master nodes in a hierarchical manner. It needs an extremely accurate time to be provided by the master node, such as a server with an atomic clock or a global positioning system (GPS). By frequently exchanging the synchronization message with specified parent nodes in the network, the leaf nodes are able

to keep synchronized with the server hierarchically. However, due to the energy limitation and network dynamics of wireless sensor networks, NTP, designed for statistic infrastructure-based network, is impractical for most of TWNs. New synchronization protocols thus are required.

In TWNs, the absolutely precise clock synchronization may not be necessary, since each node in the network carer more about the offset of its local time to its neighbors at a certain time point. Therefore the local time of a synchronization request node is usually modeled as a liner function of the reference time. Most synchronization protocols designed for TWNs aims to estimate two parameters, namely the clock skew and the offset to a reference clock, and many of them share the same design philosophies.

Generally, we can divide existing terrestrial synchronization protocols into two categories: the sender-receiver based and the receiver-receiver based approaches. In the sender-receiver based protocols, the synchronization request node (sender) initiate a two-way message exchange to the reference node (receiver). By comparing the local transmission/reception time on the pair of nodes, the sender can calculate the offset of its clock against the receiver's reference time. Both the timing-sync protocol for sensor networks (TPSN) [56] and the flooding time synchronization protocol (FTSP) [57] can be classified into this category. The synchronization accuracy of sender-receiver based approaches is affected by the uncertain of the one-way message delivery delay, including both sender and receiver errors. In receiver-receiver based methods, a third party as the reference node broadcast beacons to its neighbors. The recipients can get synchronized by exchanging their local reception time and computing the pair-wise offset against each other. Different from the send-receiver based methods, the receiver-receiver based protocols completely eliminate the effect of delay uncertainties at senders. One of the most well known receiver-receiver based protocols is reference-broadcast synchronization (RBS) [58].

Compared with wireless sensor networks, the underwater environment in UANs is more complicated. The features of long propagation delay, random drift of nodes with the ocean current and high dynamic of acoustic channel, bring grand challenges to the synchronization design. Several synchronization protocols have been proposed for UANs [31, 60, 61, 100–102] to address the unique features of UANs.

Time synchronization for high latency (TSHL) [60] is a sender-receiver based protocol, in which the effect of the skew in the high latency communication is carefully considered. To compensate the skew caused error during message exchange, a two phases synchronization is proposed. In the first phase, a reference node broadcasts beacon messages periodically letting neighbors estimate their clock skew to the reference clock. In the second phase, the sender and receiver use the skew corrected timestampts for offset measurement in the two-way message exchange. MU-Sync [61], D-Sync [100] and Mobi-Sync [31] are designed for mobile UANs. The major different among these protocols is what technique is utilized to deal with the node mobility. MU-Sync [61] is a cluster-based protocol, where the cluster head estimates the dynamic propagation delay of cluster members in round trip message exchanges. The accuracy of propagation delay estimation in this method is apparently affected by delivery delay uncertainties. In D-Sync [100] and Mobi-Sync [31], the the accurate propagation delay of the mobile node is obtained by measuring the velocity. The velocity is either estimated by measuring the Doppler shift or with the help of super node, both having specific hardware needs.

Therefore, I mainly focus on evaluating the representative synchronization protocols in static UANs, where the propagation delay is either fixed or has negligible variance compared with other delays during message exchange. The conclusions drawn in this work can be used directly to the mobile UANs regarding the message delivery uncertainties in real acoustic modems and their effect on synchronization performance.

4.2 Delay uncertainties in message delivery

The non-determinism of the message delivery delays is the direct reason for synchronization errors. Therefore, it is critical to carefully analyze and model these delays to improve the synchronization precision. The pioneers in wireless sensor networking area have extensively studied the uncertainty of delays on the radio platform [56, 57, 103]. The sources of message delivery delays are generally decomposed into six parts. They are the send time, access time, transmission time, propagation time, reception time and receive time, the details of which can be found in [56,57,103]. Among these delays, the send/receive and access time[1] , highly depending on the processor or network load, are believed to be the most nondeterministic in radios. The variation of send time and access time is in the order of tens of milliseconds. Reversely, the transmission time and propagation time have eligible variations less than 5 μs.

In UANs, so far as we know, there are no related measurements on the sources of delivery delays in real acoustic systems reported. The existing underwater synchronization protocol design and analysis are still based on the delay model proposed in radios. However, the unique physical features of the water medium and sound waveform preclude the direct use of conclusions in radios to the acoustic domain. From the tests I find that both the sources and the magnitudes of the delays in underwater message transmissions are very different from that in radio networks. Therefore, a new delay model is required for underwater synchronization design and evaluation. In this section, I give an insight into the message delivery delays in real acoustic system using UConn OFDM modems [86] as an example.

[1]Access time is the delay waiting for the channel access. Due to the narrow bandwidth in UANs, the access time is magnitudes higher than that of in radios. It can be significantly high when the network load is heavy. Since the access time is specific to the MAC protocols and can be mitigated or eliminated with careful transmission scheduling. In this work, I assume the access delay can be avoided during synchronization process in UANs.

4.2.1 Source of delays in UANs

The detailed delays involved in delivering a message are illustrated in Fig. 4.1. I use a commercial OFDM acoustic modem as an example communication device. In order to measure the magnitude of delays in each step, I made multiple time stamps during message delivery. In this figure, TS_i represents the time stamp recoded in the program on each step, and E_i is the i^{th} event involved in the message delivery. Each time stamp in my measurement has 1 μs resolution. In the tests, I just used two modems, sending and receiving synchronization messages. Therefore, the transmission of the message was neither delayed by waiting for the channel access nor interrupted by other tasks on the processor. This allows us to eliminate the mostly undetermined access time. Next, I introduce each source of delays during message delivery.

1) ***Command generation time*** *(E_1)*: the time it takes at the sender to construct the synchronization message (including sender ID, receiver ID and time stamps) on the MAC layer. This time is affected by the length of the synchronization message and the instantaneous processing load on the sender's CPU.

2) ***Command delivery time*** *(E_2)*: the time incurred delivering the synchronization from the MAC layer to the physical layer, which is the physical buffer of the acoustic modem in my example. This is mainly constrained by the writing speed of serial port.

1) ***Command detection time*** *(E_3)*: the time it takes for the modem to detect and recognize the outgoing message in the buffer. It depends on the command detection mechanism implemented in the modem, either allowing the command to trigger an interrupt or periodically scanning the buffer. The example modem used in my test implements the second periodical detection mechanism. This delay could be shorter in the interruption scheme.

4) ***Command feedback time*** *(E_4)*: the time used for the modem to send a feedback to the MAC layer informing the successful command reception. This delay is not a necessary in all operation modes of acoustic modems, thus it may not affect the message delivery delay. I have this delay measured in order to calculate the command detection time in my test.

5) ***Transmission preparation time*** *(E_5)*: the time incurred for the modem preparing packet transmission. It mainly involves the status transition delay from idle to sending, the warming up delay of the hardware device and the packet encoding delay.

6) ***Message delivery time*** *(E_6)*: the time it takes for the sender to transmit or the receiver to receive a message. This time is mostly deterministic. It depends the length of the message and the transmission rate of acoustic modems.

7) ***Signal propagation time*** *(E_7)*: the time it takes for the message to propagate in the acoustic channel. This delay closely depends on the specific application of UANs, such as the network node density, the node mobility with water current and the sound profile. The propagation delay is fixed and small in my tests.

8) ***Message reception time*** *(E_8)*: the interval between receiving the incoming message on the physical layer and sending the message up to the MAC layer. It involves the decoding time[2] and the time sending message through the serial port.

4.2.2 Measurements and analysis

To measure the sources of delays that I described in the previous subsection, I make multiple time stamps (TS_1 to TS_7) on the MAC layer, as shown in Fig. 4.1. In the message delivery delay tests, I mainly focus on the sources of delays from the hardware device. The distance between the sender and the receiver is very short to eliminate the effect of the propagation delay. A total number of 1200 packets are sent and received

[2]The decoding time is not able to be measured in this test based on the time stamps I have. It will need access to physical layer for decoding time estimation.

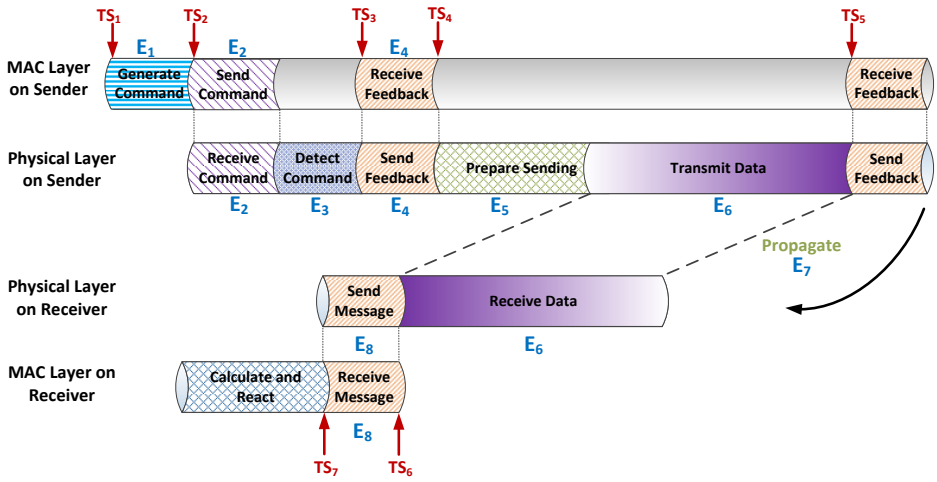


Figure 4.1: Message delivery delays in the real acoustic system.

by a pair of acoustic modems. The transmission interval between neighboring packets is 10 seconds, and the transmission power is 6 watts.

Denote the time of the i^{th} source of delays as τ_i, which can be easily calculated according to:

- $\tau_1 = TS_2 - TS_1$,
- $\tau_2 \approx \tau_8 = TS_7 - TS_6$,
- $\tau_3 = TS_3 - TS_2 - \tau_2$,
- $\tau_5 = TS_5 - TS_4 - \tau_6$,
- $\tau_8 = TS_7 - TS_6$.

A synchronization message sending in a real acoustic modem consists of a preamble block and data payload, where the preamble block is used for the automatic-gain-control (AGC) and packet detection purpose. In my test, all synchronization messages have a fixed of 38 Byte payload to eliminate the variation in transmission time. The length of preamble and data block are 490 and 320 milliseconds respectively based on the configuration of acoustic modem. Thus, the transmission time (τ_6) is about 810

milliseconds in the experiment. The reading and writing speed of the serial port, which is denoted as R_s, is $115,200$ bits per second. The lengths of packets exchanged between the MAC layer and the physical layer through the serial port in events E_2, E_4 and E_8 are $L_f^2 = 53$, $L_f^4 = 16$ and $L_f^8 = 53$ Bytes, respectively.

Denote the mean value and the standard derivation (uncertainty) of the delay τ_i as μ_i and σ_i, respectively. In addition, denote p_i^μ and p_i^σ as the percentage of μ_i and σ_i contributing on the whole delivery delay and uncertainties, where $p_i^\mu = \mu_i / \sum_{i=j}^{8} \mu_j$ and $p_i^\sigma = \sigma_i / \sum_{j=1}^{8} \sigma_j$. The values of these parameters are summarized in Table. 4.1. To give an insight into the features of these sources of delays, the temporal and statistical measurement results of each delay are shown in Fig. 4.2.

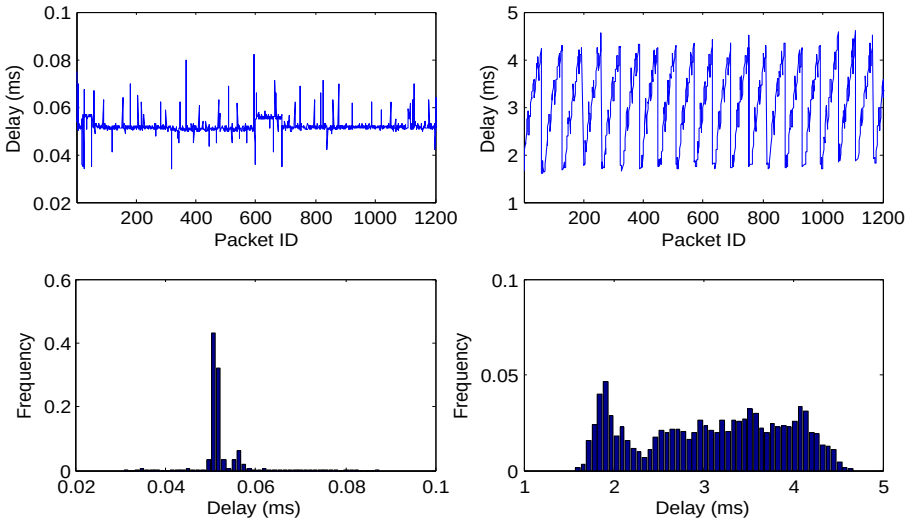
Table 4.1: The statics of delays in one-way message delivery.

E_i	μ_i (ms)	σ_i (ms)	p_i^μ (%)	p_i^σ (%)
E_1	0.04348	0.00911	0.003	0.125
E_3	3.09042	0.81294	0.245	12.212
E_5	440.96557	5.93952	34.974	81.916
E_6	810.07037	0.06821	64.248	0.940
E_7	≈ 0	≈ 0	≈ 0	≈ 0
E_2, E_8	4.06622	0.21047	0.322	2.903

From Fig. 4.2 and Table 4.1, we can draw following conclusions.

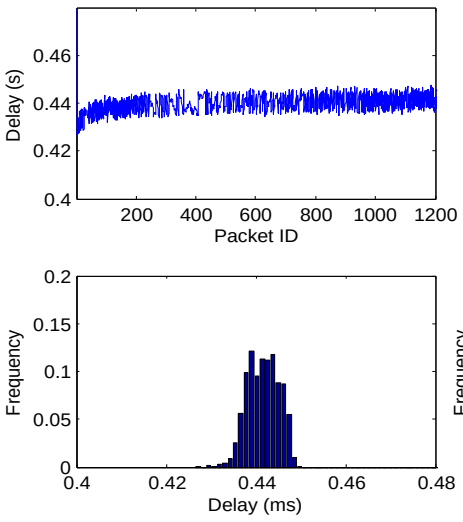
(a) The command generation time (E_1) is the most deterministic part among the sources of delays in message delivery. This delay is not only small but has very low variation compared with other delays. Its contributions on the overall message delivery delay and uncertainty are ignorable. This can be confirmed by the narrow distribution of command generation time in Fig.2(a).

(b) The command detection time (E_3) in the OFDM modem is usually higher than 3 milliseconds. In addition, the uncertainty of E_3 is about 0.813 millisecond,

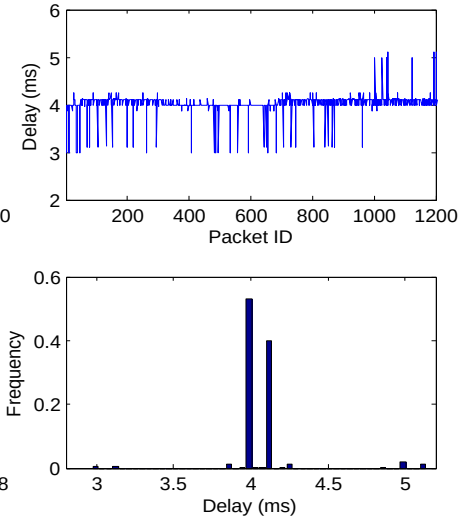


(a) Command generation time (E_1)

(b) Command detection time (E_3)



(c) Transmission preparation time (E_5)



(d) Message reception time (E_8)

Figure 4.2: The temporal and statistical results of the synchronization message delivery delays in real systems.

contributed more than 12% on the overall uncertainty of the message delivery delays. Therefore, the effect of command detection delay on the synchronization performance is considerable. This is partially resulted from the periodical command scanning design in the OFDM modem. As shown in Fig. 2(b), the command detection time (E_3) periodically changes about every 60 packets, i.e. 10 minutes (the interval between each packet is 10 seconds). An explanation about this periodical variation is the different ticking rates between the oscillator of host computer generating the message and the oscillator pulsing on the acoustic modem. Specifically, assuming the acoustic modem scan the command buffer every T_d milliseconds, and the clocks at the MAC layer and at the physical layer have $\triangle s$ skew, then the cycle of E_3 will be $P_{e3} = T_d/\triangle s$. Due to triangle wave shaped E_3 in time domain, it follows an approximately uniform distribution and has large variance. Therefore, the command detection delay becomes a significant source of error in time synchronization for UANs.

(c) The message delivering between the MAC layer and physical layer depends on the length of message and the baud rate of the serial port. In real sea experiments, the baud rate varies from 38,400 to 115,200 depending on different scenarios. I used 115,200 as an example. The mean value of this delay in my tests (E_2, E_8) is about 4 milliseconds, with variance around 0.23 milliseconds. Compared with command detection time and transmission preparation time on acoustic modems, this delay is much less significant. Therefore, we can model E_2 and E_8[3] as a deterministic delay. Its value can be computed based on message length and baud rate of serial port.

(d) The transmission preparation time (E_5) in the acoustic modem is one of the most significant delays for message delivery. It includes the time incurred for status

[3] $\tau_2, \tau_8 \approx L_f^8/R_s = 8\times53$ Bytes/115200 bps

transition from idle to sending and amplifier warming up for transmission. This delay is hardware related, which is thousands times higher than that of on a radio platform, such as Berkeley Mica2 mote [104]. Moreover, E_5 has the highest variance, contributing almost 82% on the overall uncertainty of the message delivery delays. Therefore, E_5 is the most nondeterministic part and the major source of error in synchronization. However, its well-behaved Gaussian distribution, as shown in Fig. 2(c), enables us to significantly reduce the error statistically.

(e) Due to the long preamble and the low transmission rate of acoustic modems, the message delivery time (E_6) in UANs is one of the dominant delays in message delivery. However, both the statistics in Table. 4.1 and the distribution in Fig.2(d) reveal a fact that the transmission delay is one of the most deterministic parts among the delivery delays. According to the small variation of E_6, its contribution on the overall uncertainty of the message delivery delays is less than 1%. Therefore, we can consider E_6 as a constant, and measure it at the beginning of a synchronization protocol.

To summarize, the message delivery delay in acoustic communications are apparently different with the radio communications, not only on the delay magnitude but on the source of delays. The command generation time, message transmission time and receive time are deterministic and can be measured or estimated before synchronization. In contrast, the command detection time and transmission preparation time have high uncertainties, accounting for the major sources of errors in synchronization.

4.3 Synchronization algorithms studied

In this section, I first introduce the key features of the representative synchronization algorithms studied via experiment results. The protocols were carefully implemented according to their original design. However, I took some appropriate modification during implementing each protocol, which are also described in this section.

4.3.1 Sender-receiver based synchronization

The sender-receiver synchronization scheme operates with two-way message exchange between a sender and a receiver. The representative sender-receiver synchronization protocol I tested uses TSHL [60] as a prototype and I call it as TSHL for real systems, shorted for TSHL-RS.

TSHL is a time synchronization protocol designed for high-latency underwater networks, addressing the long propagation delays. The basic mechanism of TSHL is to first model the clock skew in Phase 1, and measure the clock offset with a two-way message exchange after skew-correction in the second phase. The procedure of TSHL is shown in Fig. 4.3. In Phase 1, a reference node broadcasts beacon messages periodically letting the nodes within communication range to estimate their clock skew with linear regression. The estimated skew helps provide sensor nodes skew corrected local time in Phase 2. In the second phase, TSHL works similarly to the classical hand-shake based synchronization protocols, such as TPSN. The distinction is that all time stamps in TSHL are skew compensated to eliminate the skew effect in the high latency network.

Implementation details of TSHL-RS In this pair-wise synchronization method, I did not use a third party as reference node in Phase 1. Instead, the receiver (providing reference time) periodically send beacon messages to the sender, allowing the sender node to estimate its relative clock skew. In the network, a node will maintain a skew

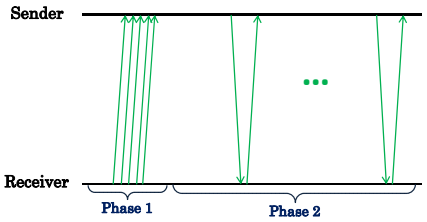


Figure 4.3: Scheduling of TSHL-RS.

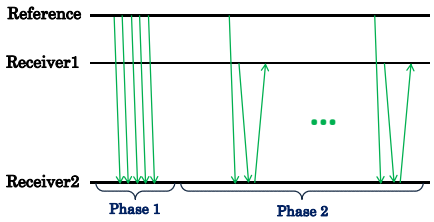


Figure 4.4: Scheduling of RBS-UW.

table listing the relative skews in difference synchronization pairs. The reason to make this modification is that there might not be always practical to achieve a consensus on a global reference in the network. After receiving M beacon message, the sender (requesting synchronization) sends out a phase switch signal and the pair of nodes transit to Phase 2. In this phase, the sender records the skew compensated synchronization request time (T_1) and reply reception time (T_4) to calculate the clock offset. Instead of carrying request reception time (T_2) and reply sending time (T_3) in the synchronization reply message, the receiver includes only $(T_2 + T_3)/2$ to reduce the packet length and lower down the errors in synchronization.

$$T_{offset\_SR} = (T_1 + T_4)/2 - (T_2 + T_3)/2. \tag{4.1}$$

4.3.2 Receiver-receiver based synchronization

The basic mechanism of receiver-receiver based synchronization is using a broadcast references to synchronize a set of receivers using Equation (4.2). The clock offset can be estimated by comparing the difference between the reception time on the receivers. The advantage of receiver-receiver synchronization methods is the capability of eliminating the sender side uncertainties. This synchronization scheme is based on the assumption about negligible propagation delays, which does not hold in UANs. Thus the receiver-receiver based synchronization schemes has been suggested unapplicable in high latency networks. However, this synchronization scheme will work as long as the nodes in the

network has the pair-wise propagation delay information. As underwater nodes are usually fixed to anchors or settled to the seafloor, it becomes a reasonable assumption of measuring the distance with relative high precision. Therefore, I would like to evaluate the performance of receiver-receiver based synchronization in UANs.

$$T_{offset\_RR} = T_1 - T_2. \tag{4.2}$$

The representative receiver-receiver based synchronization algorithm I choose to study is RBS, a well known time synchronization protocol in radios. In RBS, reference nodes periodically broadcast beacons to their neighbors. The receivers exchange the time of message reception as point of reference to estimate their clock offsets. The precision of measurement is improved statistically with multiple observations. With least-square linear regression performed, each receiver estimates its clock skew to the reference node.

Implementation details of RBS-UW To make RBS applicable in high-latency acoustic networks, I have made a number of modification as listed below. Since the new version is modified for underwater environment, I name it as RBS-UW.

- Assume the propagation delay between two receivers can be measured before synchronization. The propagation delay estimation error can be added up to the synchronization error. But in this work, I do not consider the propagation delay as a source of error but focusing on the error from acoustic systems. Improving the accuracy on measuring propagation delay is beyond the scope of discussion in my work.
- To eliminate the clock skew effect in the high latency network, I measure the clock skew periodically as designed in TSHL. That means a skew estimation phase is added before the RBS synchronization and use skew corrected time stamps (on MAC layer)

in the synchronization phase. The number of beacon messages broadcasted is the same as I use in TSHL-RS, as listed in Table 4.3.

- To prevent collision between message exchange on two receivers, I let the reference node schedule the sending orders for the receivers. The receiver scheduled later "replies" only after receiving the time stamp from the former receiver. Timeout and retransmission mechanisms are implemented to make sure successful message exchange.

4.3.3 Implementation of resynchronization

There are two major reasons for the need of resynchronization. The first one is the error in clock skew estimation. Though we can increase the accuracy by broadcasting more beacon messages (as listed in Table 4.2), this improvement is at the cost of more energy and bandwidth consumption. In UANs, the underwater nodes are usually powered by batteries making energy efficiency critical. Meanwhile, the bandwidth of acoustic communications is considerably limited than the radio networks. There is a trade-off between skew measuring accuracy and overhead. The skew estimation errors will drive the synchronized clocks apart with the elapsed time after synchronization. Offset measurements in periodical resynchronization can make up the growing error caused by the imperfect skew estimation. The second reason is accounting for the high uncertainties of packet delivery delays as discussed in Section 4.2. A single synchronization process has high error for both sender-receiver and receiver-receiver algorithms. We can therefore improve the synchronization precision statistically, by using multiple synchronization samples.

Trading accuracy with energy efficiency, I choose to broadcast 60 beacons in Phase 1 for skew estimation. In the experiment, I repeated the synchronization process in Phase 2 every T interval. The implementation parameters are listed in Table 4.3.

Table 4.2: Clock skew error in linear regression.

Number of beacons	25	40	60	80	200
Mean ($\mu s/s$)	7.63	3.28	2.14	1.22	0.27
Standard variation ($\mu s/s$)	5.66	2.81	1.71	1.08	0.20

Table 4.3: The statics of delays in one-way message delivery.

Number of beacon messages (Phase 1)	60
Beacon sending interval (Phase 1)	10 seconds
Periodic synchronization interval (Phase 2)	10 minutes
Clock skew re-estimation interval	5 hours

As the clock skew among nodes might slowly change with environmental factors, such as temperature and supply voltage. Without a correction of clock skew, the synchronization accuracy decays as more time elapse after the preceding skew estimation. Putting this into perspective, I repeat the clock skew estimation phase (Phase 1) every 5 hours for all protocols in the experiments.

4.4 Experiment evaluation

In this section, I test and compare the performance of representative time synchronization protocols, TSHL-RS and RBS-UW, in a lab environment. Each underwater node was consisted of an acoustic modem [86] and a controller launching the protocol stack. The synchronization protocols, TSHL-RS and RBS-UW, runs on the MAC layer. The modems were placed closely in a water-tank, so the propagation delay was negligible.

The random variations of one-way message delivery delay contributes directly to the synchronization error in sender-receiver based algorithms, while the nondeterminism of the receiving time being a major source of error in receiver-receiver based algorithms.

Therefore, before embarking on a discussion of how time synchronization protocols perform in real systems, it is critical to first evaluate the uncertainties at the sender and receiver sides and how each part of delays introduced in Section 4.2.1 contributes in the error.

4.4.1 Characterizing the errors

When measuring the one-way delivery delay uncertainty, I connected two acoustic nodes to a common host computer, which provides a reference time clock. One node played the sender and initiated the message exchange. The message sending and receiving time were recorded at both nodes. Since I have a common reference clock, the time stamps can be converted to the reference clock for a fair measurement. I calculate the one-way delivery differences on each message exchange for 3000 runs. The magnitude of the one-way delay variation is shown in Fig. 4.6, which gives us a cue on the synchronization error for sender-receiver methods in real acoustic systems. The distribution of this uncertainty is clearly Gaussian, with standard variation 7.254 milliseconds. The well-behaved distribution will enable us to improve on the synchronization precision statistically through multiple synchronization rounds. This enhancement will be discussed in Section 4.4.2.

In the set up for receiving delay uncertainty, I used one node periodically broadcasting messages to two receiving nodes. The two receivers were connected to a host computer for clock reference. In the test, I recorded the receiving time at both receivers and converted the timestamps to the common reference clock. The difference of the reception time on two nodes accounts for the receiver uncertainties, similar to the measurement in radios [58]. The effect of propagation delay was eliminated in my test, as the three nodes were placed very close in a water tank. This experiment tests the

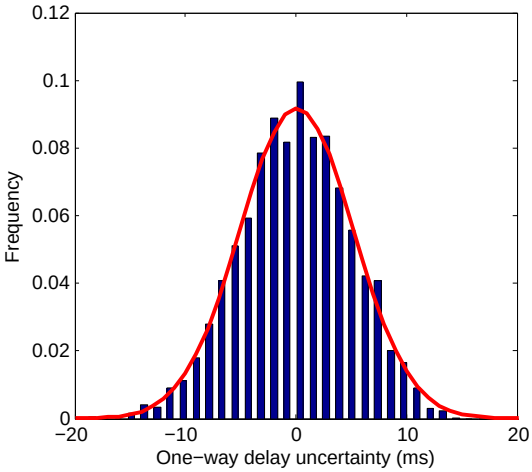


Figure 4.5: A histogram showing the distribution of one-way message delivery delay uncertainty in the real acoustic system. The curve is a plot of the best fit Gaussian fitting. (μ=0.225 msec, σ=7.254 msec)

uncertainty on receiving packets in real acoustic systems. The receiving time variation in 3000 runs are displayed in Fig. 4.6.

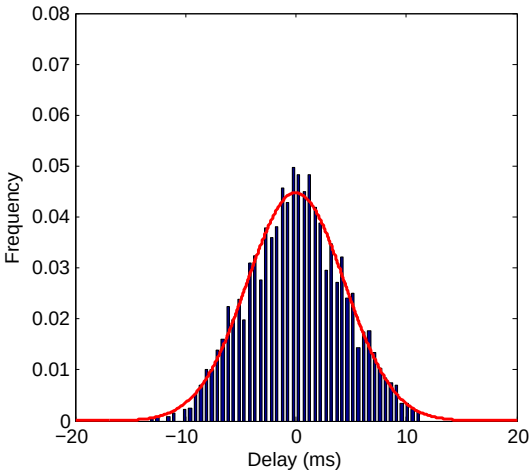


Figure 4.6: A histogram showing the distribution of reception time difference in the real acoustic system. The area within two vertical lines is 0.90. (μ=0.54 μs, σ=6.247 msec.)

The receiver errors also have a Gaussian distribution, but with lower variation in magnitude than the one-way delivery delays. Lower variance means less error in a single synchronization process. Moreover, the receiver error has nearly zero mean while the

average one-way delivery delay is about 2.5 milliseconds. This indicates RBS-UW can theoretically achieves high precision synchronization (around 1 μs error or less) with more synchronization counts. However, the statistical method cannot exclude all errors in TSHL-RS, as the one-way delivery delay uncertainty is not zero mean, which limits its synchronization accuracy in real UANs. However, when the receiver-receiver based synchronization methods are applied in real UANs, where the propagation delay is significant, the error in reassuring pair-wise propagation delay can be added up to the synchronization error.

4.4.2 Performance evaluation

In the previous subsection, I measured the uncertainties of one-way message delivery delay and the receiving time, which directly contribute to the synchronization errors in sender-receiver and receiver-receiver based synchronization algorithms, respectively. In this subsection, I test and compare the representative protocols, TSHL-RS and RBS-UW in real systems. In the tests, I connected the nodes to one common computer in order to get a reference clock for synchronization error measurement. The tested protocols were implemented as described in Section 4.3 and the synchronization errors were compared to evaluate their performance in UANs.

Fig. 4.7 displays the synchronization error of TSHL-RS in a 10 hour test. The sender estimated the clock skew to the receiver every 5 hours with 60 broadcasting beacons. Fig. 4.7 shows the synchronization error in Phase 2. The blue curve presents the error of each single synchronization process, the interval of which is 10 minutes in Phase 2. The red curve shows the statistically enhanced TSHL-RS with linear regression. When there is no clock skew between sender and receiver, or zero error for skew estimation, we can simply improves the precision by averaging up multiple synchronization processes and show the result as black curve in Fig. 4.7.

Due to the high uncertainty of one-way message delivery time, each single synchronization attempt has large and highly dynamic errors (blue curve in Fig. 4.7). The magnitude of single synchronization errors is consistent with the one-way delivery delay uncertainty. Recall from Table 4.2 that the skew estimation has about $2\mu s/s$ with 60 beacons, the skew estimation error would cumulatively affect the offset measurement. Therefore, I let the sender refine the skew measurement with the received message in Phase 2 and use linear regression to improve the offset calculation. We can see from Fig. 4.7, the synchronization error can be apparently reduced comparing to the single measurement. This enhancement highly depends on the sample size of linear regression or more essentially the accuracy of skew estimation. Hence in Fig. 4.7, I also present the performance of TSHL-RS eliminating the clock drift between sender and receiver. We are able to do that since both nodes were connected to a common reference computer for clock correction. With simply averaging the results from multiple measurement, the error can be significantly reduced to lower than 2 ms.

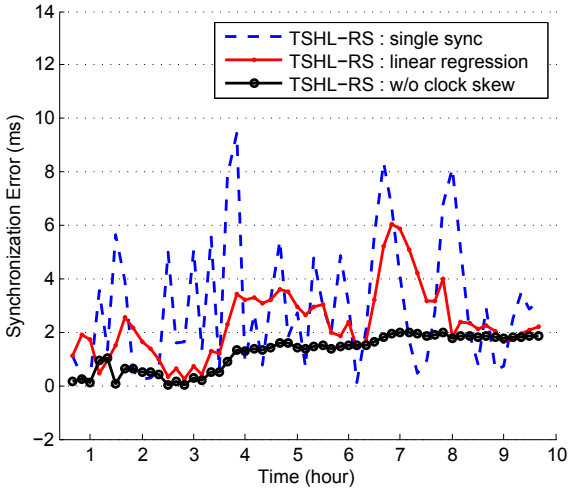


Figure 4.7: Comparison of errors for sender-receiver synchronization.

The results of RBS-UW synchronization in real acoustic systems is shown in Fig. 4.8. In the 10 hour test, the two receivers similarly calculate their clock skews to the reference node every 5 hours. The synchronization proceed after skew estimation.

Due to the high uncertainty on the receiving time as shown in Fig. 4.6, RBS-UW also has highly varied synchronization errors in each single synchronization procedure. Given the skew estimation errors, the offsets measured by multiple reference packets monotonously increase with time. Therefore, we are not able to improve the precision as suggested in [58] by taking the average of multiple measurements. To reduce the synchronization error of RBS-UW, I utilize linear regression to 1) correct the skew estimation error and 2) improve the precision of offset measurement. The synchronization error was significantly reduced to less than 2 ms with the help of linear regression, especially when more samples were available in the second half of experiment. Note that the red curve also has high variation at the beginning of the test due to the lack of samples in linear regression. This can be improved with more frequent synchronization attempts in the first hour of test. For comparison purpose, I show the performance of RBS-UW when eliminating the clock skew effect. As we can see from black curve, it has comparable stable accuracy to the linear regression enhanced RBS-UW, but can lower down the error much faster. I observe similar phenomenon in testing TSHL-RS, as shown in Fig. 4.7.

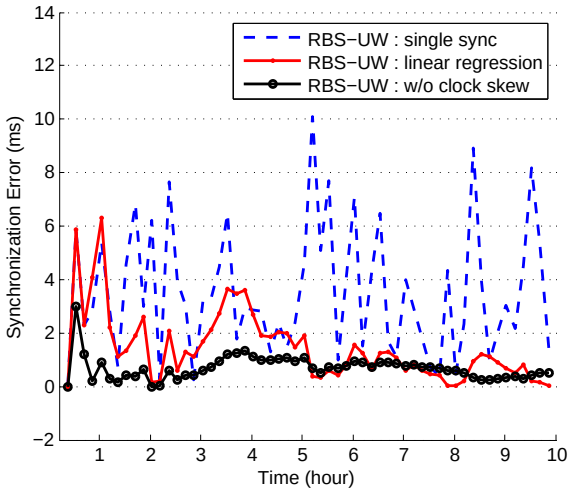


Figure 4.8: Comparison of errors for receiver-receiver synchronization.

To summarize, the high uncertainties in message delivery delay, especially the command detection time and transmission preparation time in acoustic modems, significantly increase the synchronization error in UANs. To compensate both skew and offset estimation errors, periodical synchronization is required in real applications. According to the experiment results, the linear regression in synchronization phase can effectively on compensate the skew estimation error and reduce the effect of delay uncertainties on offset measurement. RBS-UW outperforms TSHL-RS slightly on compensating delay uncertainties, as it is free from the sender uncertainties. However, the error in propagation delay measurement will potentially degrade the performance of RBS-UW in a real network.

4.5 Summary

In this chapter, I identified the source of message delivery delays in real acoustic systems. Due to the fundamental difference between acoustic and radio communication systems, the delay modem in UANs have distinct features. Specifically, the significant command detection delay and transmission preparation delay are highly nondeterminism and become the major source of errors in time synchronization. I also characterized the one-way message delivery uncertainty and the receiver uncertainty in UANs, which essentially contribute the synchronization error in sender-receiver and receiver-receiver based synchronization algorithms. Based on my observation, the distribution of both uncertainties are Gaussian shaped, but have significantly higher magnitude that that of in radio systems. The high one-way delivery delay and receiving delay uncertainties resulted high synchronization errors in real systems, which is consistent with my experiment results in performance evaluation on sender-receiver based TSHL-RS and

receiver-receiver base RBS-UW. RBS-UW slightly outperforms TSHL-RS on compensating the delay uncertainties, as the receiver-receiver based design completely eliminates the sender uncertainty. However, the errors on propagation delay measurement would degrade the performance of RBS-UW in real applications. How TSHL-RS and RBS-UW perform in real field experiments with long propagation delay will be my future work.

Chapter 5

Impact on Underwater Security

As the source of life, the oceans never stop attracting people's attention in both academia and industry. UANs enable scalable and distributed data acquisitions in a wide spectrum of applications, including unmanned ocean exploration, ocean surveillance and deep water oil drill protection. The possibility of secure message delivery may determine the success or failure of a mission. Therefore, how to secure the communications in UANs is becoming an important topic.

Like terrestrial sensor networks, UANs are susceptible to various attacks, which target different components in the system. For example, attacks like wormhole target at routing protocols [62], and jamming attacks can disrupt links between nodes [63, 64]. An adversary can also violate communication security by passively eavesdropping the private signal or actively injecting fake information to the network [66]. Among the aforementioned security issues, the communication security is one of the most fundamental and critical tasks in underwater networks, which use broadcast channel for acoustic transmissions. The public-key cryptography are nearly infeasible in the networks with constrained energy and processing power [67]. Alternatively, symmetric-key ciphers are often used to provide confidentiality in underwater communications because of their performance advantages [68, 69].

However, symmetric-key cryptography require a shared secret key by sender and receiver for both encryption and decryption. The requirement that both parties have access the the secret key makes the key generation and key exchange challenging, especially in resource constrained UANs. It is difficult, if not impossible, to specify an online KDC in oceans to allocate secrete keys among devices. The most accepted solution is a combination of pseudorandom key generators and key predistribution [68, 69]. However, lack of randomness in those generators is a common problem leading to cryptanalytic breaks. Key predistribution have connectivity and resiliency issues. An isolated node possibly exists when it has no common key with neighboring nodes. All the methods that preinstall keys on the nodes also have the risk that a single compromised node might result in a number of unsafe parties sharing common keys with the compromised node.

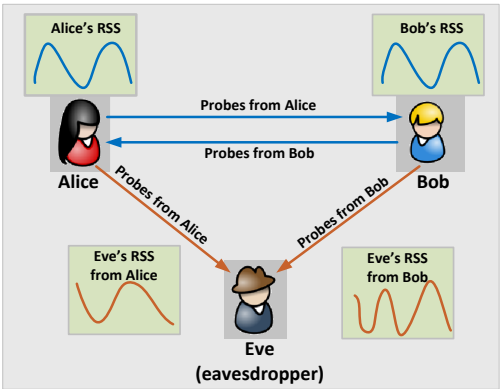


Figure 5.1: RSS measurements in the network.

RSS-based key generation schemes [70–73], however, allows each pair of nodes, after being deployed, to update secret keys easily at any time. In RSS-based key generation schemes, the randomness of the keys depends on the entropy naturally available in the environments. The communicating parties on the two ends of a reciprocal link can produce a shared key through local RSS measurements [74]. An opponent that is monitoring the communication channel, however, can hardly guess the secret key if it

is physically near neither communicating entities [76]. This security is consequently ensured with the spatial diversity of acoustic channel, as shown in Fig. 5.1.

In this chapter, I explore the advantages and challenges of RSS based key generation in UANs, and meanwhile, evaluate representative RSS based key generation methods in sea tests. The content in this chapter is partially based on my previous work published in [78][1] .

5.1 Background and related work

Most of earlier RSS based secret key generation approaches did not involve the signal preprocessing mechanisms into concern. They were mainly focus on improving the key extraction method.

The authors in [105] proposed to use the envelope of receiving power as the RSS sequence to extract a key. In order to reduce the bit mismatch rate, samples outside two predetermined thresholds are discarded. In [106], the correlated observations of deep-fading by two parties that wish to secure communications is utilized to generate the key. In this work, only one threshold need to be determined through the model of the channel. Any measurement below this threshold is considered as deep-fading response, and will be quantized to a bit "0". However, compared with the non-fading time in a channel, the duration of deep-fading is much shorter. To avoid the low zero-one rate of the key, the universal hash families thus is used to increase the extracting randomness. An approach that use the channel impulse response to produce a key is studied in [107]. A high key generation rate can be excepted with this approach, since each probe signal can generate plural RSS measurements in a multipath environment. However if Eve get some basic information, like the distance and the depth, of Alice

[1] © 2016 IEEE. Reprinted with permission from Yu Luo and Lina Pu, RSS-based secret key generation in underwater acoustic networks: advantages, challenges and performance improvements, IEEE Communications Magazine, Feb. 2016 (Accepted)

and Bob, the multipath response between Alice and Bob may be modeled by her through these geometry knowledge [108, 109]. In addition, compared with the SNR, the multipath response may be stable in a long period, which causes two keys produced by different probes are highly correlated.

Essentially, the aforementioned key generation approaches used similar raw data structure. The mismatch rate and the randomness of the key may be greatly affected by the noise, asynchronized probe transmission and the large-scale fading of channel. To address these problems, the signal preprocessing process thus is required. The generally used preprocessing approaches include directional transmission and reception [70], deep-fading elimination [71], interpolation and decorrelation [72], which will be introduced in Section 5.3. By using these approaches, both the correlation among neighboring measurements and the RSS sequences disagreement between two key generation parties can be reduced.

Some post-processing methods can also help to refine the key after key extraction. In [110], the information reconciliation and the privacy amplification are employed to decrease the bit mismatch rate and the leakage rate of a secret bit stream. In the information reconciliation stage, Alice and Bob exchange error-correcting messages over an authenticated public channel, which allows them to agree on an identical bits string. However, this procedure may incur information leakage to the adversary Eve, and rely on the assumption on the existence of a secure public channel. The authors in [111] presented a reconciliation protocol, which leaks an amount of information acceptably close to the minimum possible for sufficiently reliable secret channels. In privacy amplification [112], Alice and Bob remove a part of information revealed to Eve to guarantee that adversary cannot use this information to guess the secret key. The drawback of privacy amplification is that it reduces the size of the secret bit stream which decreases the generation rate of a key.

In the remaining section, the key features of Aono's, Mathur's and Patwari's secret key generation approaches studied in the experiments are briefly introduced. All three approaches were carefully implemented according to their specifications published as of Mar 2014 and based on clarifications of some issues from the designers and on my own experimentation with them.

Here, I segment each approach, if possible, into two independent stages, which are the *signal preprocessing* and the *key extraction*. In particular, the signal preprocessing helps Alice and Bob to produce an appropriate input sequences based on the original RSS measurements, while the key extraction quantizes the output of signal preprocessing into the secret bit streams. According to the extraction methods, the key generation can generally be classified into two categories: (a) the single-bit approaches, which quantize each measurement point to at most one bit, and (b) the multi-bit approaches, which extract multiple secret bits from a single RSS measurement.

5.1.1 Aono's key generation

Aono [70] is a single-bit key generation approach. It requires Alice or Bob to equip a smart antenna for directional transmission and reception. The major advantage of Aono over other RSS based approaches is on the high diversity of channel response by using the beam-forming technique.

5.1.1.1 The signal preprocessing

In Aono, Alice and Bob do the following processes before key extraction:

a. Alice sends a series of probe signals to Bob through a smart antenna. Each probe is transmitted on a different direction to increase the diversity of the channel response.

b. Once Bob received all probes from Alice, he replies the same number of probes to Alice through an omni-directional antenna.

c. Alice switches to the receiving mode. The receiving direction of each probe is the same as the one she used for transmitting the corresponding probe.

5.1.1.2 The key extraction

After the signal preprocessing, Alice and Bob extract keys based on the RSS measurements via the following steps:

a. Alice quantizes the largest $l/2+\beta$ and the smallest $l/2+\beta$ RSS points to "1" and "0" respectively. Here, l is the length of the secret key, whose value is smaller than $N/2$, half of the total number of RSS measurements, and β is the number of redundancy bits allowing for the disagreement.

b. The identifications (IDs) of probe signals that have not been quantized by Alice in step (a) are sent to Bob. Bob removes these probes, and then quantizes the largest $l/2$ and the smallest $l/2$ measurements from the rest of RSS points to "1" and "0" respectively. The IDs of the unchosen probes are also sent back to Alice, who will remove the corresponding probes as well. The remainder l bits binary sequences are considered as the secret key.

c. Finally, Alice and Bob use a same error-correction coding scheme to correct the disagreements between their keys.

5.1.1.3 Implementation discussions

When testing Aono's key generation approach, I only used its key extraction mechanism and skipped its signal preprocessing stage for the following reasons: (a) due to the size constraint, most of existing acoustic modems, including the OFDM modem

I used in the experiments, have only one transducer and do not support directional transmission, and (b) the highly dynamic feature of acoustic channel [113] guarantees enough diversity of RSS sequence for key generation without using the beam-forming technique.

In the sea experiments, I found that the loss rate of probes exchanged between Alice and Bob was significantly higher than that of in territorial environments. Additionally, IDs of probe signals lost on Alice and Bob were very different. In this situation, a lot of probes which were finally selected by Alice for her key extraction might be missed at Bob, and vice versa. This extremely increased the bit mismatch rate of a key. To solve this problem, in the tests I let Alice and Bob exchange IDs of their lost probes at the end of signal preprocessing stage. After this simple but effective procedure, the intersection of successfully received probes, the number of which is denoted as N_p, was available on Alice and Bob for their key extraction.

5.1.2 Mathur's key generation

The same as Aono, Mathur [71] is also a single-bit key generation approach. In Mathur, only a group of consecutive RSS points above or below the thresholds are quantized to one bit, while other measurements are discarded. This extraction strategy can improve the bit mismatch rate between Alice and Bob with the cost of low generation rate of a key.

5.1.2.1 The signal preprocessing

In Mathur, after Alice and Bob receiving the probe sequence from each other, large-scale fading elimination is carried out to increase the randomness of the secret key. Specifically, Alice and Bob divide their RSS measurements into multiple subgroups, each of which contains u successive RSS points. The large-scale fading is removed by

subtracting each RSS value with the average of corresponding subgroup. After this process, only the small-scale fading is left for the following key extraction.

5.1.2.2 The key extraction

After the signal preprocessing, Alice and Bob extract their secret keys with the following steps:

a. Set up the thresholds q_+ and q_- based on the RSS sequence, where $q_\pm = \mu \pm \alpha \cdot \sigma$, μ and σ are the average and the standard deviation of RSS sequence respectively, and α is a quantizer level coefficient.

b. Alice searches the positions of all *excursions*, where an excursion is defined as m successive RSS points above q_+ or below q_-. Then Alice sends the ID of middle prob of each excursion to Bob.

c. Bob identifies the intersection of excursions between Alice and himself, and then sends the positions of these intersected excursions back to Alice.

d. After Alice getting the feedback from Bob, she drops excursions which are not overlapped with Bob from consideration.

d. Finally, each excursion is quantized to a single bit "1" if its RSS measurements are above q_+, or "0" if the ones are below q_-.

5.1.2.3 Implementation discussions

Unlike Aono, the lost probes do not affect the bit mismatch rate in Mathur, since the steps (b) to (d) in its key extraction stage guarantee Alice and Bob to use common RSS points for their key generation. This is one of the advantages of Mathur.

The specification of Mathur [71] does not explicitly define the value of u (the number of RSS points in each subgroup for large-scale fading elimination). From experiments

I found that the parameter u cannot be too large or too small. A large u may not be able to effectively remove the large-scale fading effect, while a small u tends to erase too many details of the small-scale fading, both of which decrease the randomness of a key. Additionally, in experiments I observed that the duration of large-scale fading was not constant. Therefore, although I used a fixed u in my tests, changing this parameter adaptively according to sea conditions may help to further improve the randomness of the key.

In Mathur, there is a trade-off between the key generation rate and the bit mismatch rate by adjusting the amount of RSS points, m, in the excursion. A larger m decreases the key generation rate but improves the bit mismatch rate, and vice versa. However a "noisy" underwater acoustic channel [113] makes the RSS data fluctuate quickly. In this situation, if we use a large m as used in radio environments, the key generation rate will be very low. This implies that Alice and Bob will need to send a larger number of probes to generate a key of same length. In order to balance the performance between bit mismatch rate and key generation rate, in the experiment I scanned different values of m, and chose an appropriate one for Mathur.

5.1.3 Patwari's key generation

Patwari [72] is a multi-bit key generation method. By uniformly segmenting the cumulative distribution of RSS into multiple intervals, each measurement point in its corresponding interval can be quantized to plural bits. Additionally, Patwari takes the problems of asynchronous transmission of probe signals between Alice and Bob, and the high correlation among neighboring RSS measurements into account to improve the bit mismatch rate and the randomness of the secret key.

5.1.3.1 The signal preprocessing

In Patwari, the raw RSS sequences measured by Alice and Bob go through a p-order Farrow fractional interpolation filter [114] first to reduce the disagreement of keys caused by the asynchronous probe transmissions between Alice and Bob in half-duplex mode. Compared with the original RSS measurements, the reciprocity of the output is considerably improved. After the interpolation, a Karhunen-Loève decorrelation transformation (KLT) [115, 116] is utilized to reduce the correlation of the RSS sequences before key extraction.

5.1.3.2 The key extraction

After the signal preprocessing, the following steps are executed on Alice and Bob to extract the secret keys:

a. Divide the cumulative distribution of KLT's output into 4×2^k intervals, where k is the number of bits used to quantize each data point. The ID of interval i is denoted as m_i, $i = 1, 2, \ldots, 4\times 2^k$.

b. Both Alice and Bob generate two sets of k-bits Gray codes, and each codeword is repeated four times. The codeword i in the set j is denoted as $d_j(i)$, where $j=0, 1$ and $i=1, 2, \ldots, 4\times 2^k$, while set d_0 is the circularly shift of set d_1[2] .

c. Alice gets the binary value e_i for her data points based on the interval ID of the data, and then sends the $\mathbf{E} = [e_1, e_2, \ldots]$ vector to Bob.

d. After Bob receiving the $\mathbf{E}$ vector , both Alice and Bob encode their secret keys with codeword d_1 whenever $e=1$, or with codeword d_0 whenever $e=0$.

[2] $d_0(l)=d_1(a\%b)$, where % is the remainder operation, $a=l+2$ and $b=4\times 2^k+1$.

5.1.3.3 Implementation discussions

In Patwari, if the positions of lost probes are different between Alice and Bob, the bit mismatch rate of a secret key will be very high. The reason is that Alice and Bob in Patwari do not exchange the information on the prob IDs they use to produce secret keys. Therefore, the keys produced by Alice and Bob may be "maloccluded" with each other in the case of high probe loss rate. For example, I assume that Alice and Bob send four probes, and each measure point is quantized to two bits. Suppose the ideal key is 10 01 11 00. However, due to an imperfect reception, assuming Alice loses the first probe and Bob loses the fourth one, the keys generated by Alice and Bob are 01 11 00 and 10 01 11, respectively. It is easy to observe that five out of six bits are different between in the two keys. To mitigate the negative impact of lost probe on bit mismatch rate, when implementing Patwari, I use the same method to exclude the lost probes as applied in Aono (Section 5.1.1.3).

Moreover, from experiments I found that the performance of Patwari was sensitive to the parameter k, the number of bits used to encode each measurement point. Although a large k could improve the key generation rate, the bit mismatch rate will also increase significantly, as the last several bits of each encoded RSS measurement will be dominated by the asymmetric noise. Since the best value of k is not explicit specified by authors and other researches [72], I scanned different values of k in sea tests, and chose an appropriate one to balance the performance between bit mismatch rate and key generation rate for Patwari.

Table 5.1 lists the parameters I used for Aono, Mathur and Patwari in experiments.

5.2 Advantages and challenges

In RSS-based key generation schemes, the randomness of the keys depends on the entropy naturally available in the environments. The shared key can hardly be guessed

Table 5.1: Parameters used in sea tests

Aono	
Length of the key (l)	$0.1 \times N_p$
Number of redundancy bits (β)	$0.5 \times l$
Mathur	
Number of points in each subgroup (u)	50
Quantizer level coefficient (α)	0.1
Number of RSS values in each excursion (m)	2
Patwari	
Number of bits for each quantization (k)	3
Order of Farrow filter (p)	4

if an opponent is physically near neither communicating entities as a consequence of the spatial diversity of acoustic channel. In this section, I discuss the benefit of RSS key generation and the challenges from the underwater environment.

5.2.1 Advantages of RSS key generation

Compared to conventional cryptographic key generation schemes, the RSS-based methods have the following advantages, which make them promising techniques for UANs.

- *Feasibility*: Any two parties that want to communicate secretly can simply use a point-to-point probe transmission protocol to generate a key without the participation of any key management entities. In addition, as a critical parameter in communication systems, the RSS could be measured by most of commercial acoustic modems directly without any modification on the hardware or on the software.

- *Security*: Unlike pseudorandom key generations, which have potential cryptanalytic breaks in large networks, the security of RSS-based methods is naturally preserved by the spatial diversity and random variation of acoustic channel. Particularly, an attacker close to neither communication entities will measure an uncorrelated

channel, and thus hardly guess the key through overhearing. Furthermore, the high dynamic of acoustic channel guarantees that the RSS sequences collected in different time periods are uncorrelated [113], which is a favorable feature allowing a pair of nodes to flexibly update their secret key at any time.

- *Resilience*: RSS-based key generation schemes have high resilience, since the compromise of some good nodes will definitely not reveal the security information of other links in the network. The secret keys are essentially produced from local measurements on the channel response, which have significant diversity among different links. A pairwise key for two communication parties is unknown to any other entities. The high resilience of RSS-based key generations provide good quality of resistance against the hacking attempts to the network.

- *Scalability*: Apart from conventional pairwise key sharing schemes which requires large memory to store a considerable amount of preinstalled keys in large scale networks, the RSS-based key generations have no constraint on the memory space. Different and random keys are naturally created benefiting from the entropy feature of underwater environments. Therefore, the RSS-based scheme could operate efficiently in the large size UAN or in networks with incremental deployments.

- *Key connectivity*: Key connectivity represents the probability that two neighboring nodes have common keys to establish a secure link for communications. A high connectivity requires a large amount of shared keys on two nodes in conventional random key generators. The requirements on resilience, scalability and connectivity, however, are conflicting in general symmetric key generation approaches. In RSS-based key generation schemes, any pair of nodes can produce shared keys as long as two nodes are physically reachable through acoustic channel. Upper layer services like routing will get considerable benefits from the high connectivity of the key in RSS-based key generation schemes.

It is worth noting that the RSS-based key generation approaches mainly focus on protecting the communication between the authenticated parties against malicious adversaries. Like other symmetric key generators, the RSS-based scheme has no mechanism for device authentication, thereby requiring to establish a initial secure link between communication entities. There have been extensive authentication mechanisms in the literature [117], which could be used in conjunction with RSS-based key generation methods. For instance, similar to Diffie-Hellman protocol, the nodes can use public-key based key exchange mechanism for device authentication. Another viable solution is to pre-distribute some temporary keys to authenticate the identities and to exchange the initial shared secret [118].

5.2.2 Challenges from UANs

Owing to the advantages introduced in the previous section, RSS based key generation has been advocated as a promising technique to provide security keys for wireless communications. A variety of RSS-based key generation approaches have been designed for wireless radio networks. However, no attempt has been made to evaluate their performance in UANs. As introduced in Chapter 2, underwater channel and acoustic communication systems have some unique features, which in turn bring grand challenges to the RSS based key generations in UANs.

5.2.2.1 Long transmission time on probe signal

Benefit ing from the wide bandwidth, the transmission time of a probe in the radio network is usually less than one millisecond. This allows the communicating entities to generate a key with desired length very fast. On the contrary, due to the long preamble signal in UANs, the transmission time of a probe could be thousands times longer than that in terrestrial environments, which causes a slow generation rate of secret keys.

More specifically, for signal detection and AGC purposes, an acoustic modem needs to attach a preamble sequence before each packet. In UANs, the length of a preamble can achieve half a second or even longer [43], 1000 times larger than that in the radio network. This prevents Alice and Bob from transmitting a sequent of probes in a short period as they do in terrestrial environments. Therefore, an RSS-based key generation approach in oceans needs a longer time to create a secret key with enough length.

Using Mathur's single-bit approach listed in Table 5.3 as an example, its key generation rate in the sea tests was 0.09 bits per probe. To produce a key of 128 secret bits, a node was required to send a total number of 1422 probes. In experiments, the minimum transmission time of a probe signal was 0.5 seconds. Therefore, two communicating entities have to take about 13 minutes for probe transmissions, which makes the single-bit approaches inefficient in UANs.

Given the long transmission time of probes in oceans, the efficiency of an RSS-based key generation becomes a major challenge on the single-bit key extraction approaches. Multi-bit key generation methods, on the other hand, can significantly improve the key generation rate, but at a cost of higher bit mismatch rate, as depicted in Tabke 5.3. How to balance the key generation rate and the bit mismatch rate in an RSS-based key generation is still an open issue in UANs.

Table 5.2: Performance comparison of three representative approaches.

Method	Signal Pre-pro. Method	Key Extraction	Par.	Key Gen. Rate	Mismatch Rate (%)	App. Entropy	P-val
Aono	Beam-forming[1]	Single-bit	$l=0.1N_p$[2]	0.10	10.8	0.34	< 0.01
Mathur	Large-scale Fading Elimination	Single-bit	$\alpha=0.1$ $n=2$	0.09	38.5	0.68	0.43
Patwari	Interpolation & Decorrelation	Multi-bit	$k=3$	3.00	49.3	0.69	0.61

[1] Due to the hardware constraint of acoustic modem, I excluded the array processing from experiments.
[2] N_p is the total number of RSS measurements.

5.2.2.2 Asymmetric RSS measurements

The RSS-based key generation relies considerably upon the reciprocity of acoustic channel. However, due to the half-duplex feature of acoustic modems and the fast variation of underwater channel, the RSS measurements on the communicating parties are not exactly symmetric, which may affect the robustness of a key generation method in terms of bit mismatch rate.

Particularly, due to the size and cost constraints, most of existing acoustic modems only equip with a single transducer for communications. These modems operate in the half-duplex mode in the sense that they are capable of either transmitting or receiving. Therefore, if two communicating parties send probes simultaneously, there may be a collision between the transmission and the reception, especially given the long transmission time of a probe signal in the underwater environment. To avoid the collisions, two parties have to use non-concurrent probe transmissions in UANs.

In radios, owing to the negligible propagation delay and the short transmission time of probes, the time difference between a pair of RSS measurements caused by the non-concurrent probe transmissions is very short, usually less than the channel correlation time. In such scenario, the effect of asymmetric RSS measurements could be mitigated by using an interpolation filter in the signal processing stage. Compared to radio networks, however, the transmission time of a probe is thousands times longer in UANs, and the propagation speed of acoustic signal in water is five orders of magnitude lower than that of electromagnetic wave in air. Hence, when sending probes non-concurrently, the time difference between the transmissions of a pair of probes in UANs is thousands of times longer than that in terrestrial environments. For this reason, the asymmetry of RSS measurements on a pair of nodes is significant in oceans.

Using a sea experiment as an example, I configured a node called as Alice being the initiator for the key generation. In each round of communication, she sent a probe signal

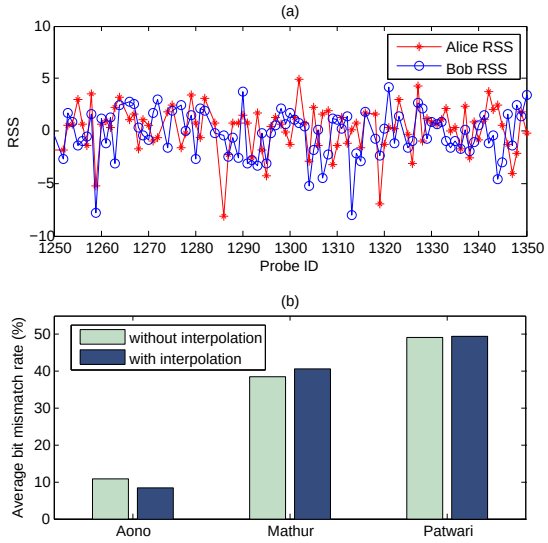


Figure 5.2: Large RSS discrepancies and high bit mismatch rates. (a) RSS measurements on two parties, (b) Bit mismatch rates with and without the interpolation filter.

to another node, Bob, who replied the same signal after he received the probe from Alice successfully. This procedure was repeated multiple times until two parties got enough RSS measurements for the key extraction. The distance between two communicating entities in the experiment was 556 meters, therefore the minimal time difference between a pair of RSS measured by Alice and Bob was $t_p + t_s = 0.87$ seconds, where t_p is the propagation delay (0.37 seconds) and t_s is the probe transmission time (0.5 seconds).

Fig. 5.2 (a) demonstrates the asymmetric RSS measurements caused by the non-concurrent transmission of probe signals in oceans. This finally results in a high bit mismatch rate on the RSS-based key generation approaches, as listed in Table 5.3. In addition, this bit mismatch rate cannot be evidentially reduced by applying an interpolation filter, as shown in Fig. 5.2 (b).

5.3 Field tests

The overall goal of the experiments was to assess the capability of the three candidate RSS based key generation approaches on producing highly random and low bit

mismatch rate keys between Alice and Bob, while avoiding Eve to accurately guess the secret key from eavesdropping. To achieve this goal, I conduced a series of experiments from November 13, 2013 to April 15, 2014 to evaluate the performance of each approach under a various of sea conditions.

5.3.1 Experiment setup

In the sea experiments, I had three nodes deployed at Long Island Sound to take the role of parties (Alice, Bob and Eve) for key generation. Each node was consisted of a surface buoy, a micro controller unit (MCU), a radio frequency (RF) modem, an UConn OFDM modem and an anchor, the details of which can be found in [119]. The UConn OFDM modem has one omnidirectional transducer and four omnidirectional hydrophones, as shown in Fig. 5.3.

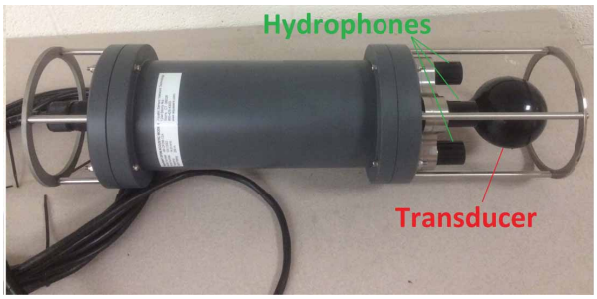


Figure 5.3: UConn OFDM acoustic modem.

The center frequency and the bandwidth of the acoustic modem are 17 kilohertz and 6 kilohertz, respectively. All modems were deployed about 20 meters below the surface, and water depth of around the experiment site was about 30 meters. The sea conditions during the experiments were collected by the MYSound real-time weather node and station [120] at Eastern Sound and Ledge Light, respectively[3] , the locations

[3]The Ledge Light weather station was much closer to the three surface nodes than the Eastern Sound station. Therefore the weather information from this station was my first choice. Eastern Sound monitoring node was a backup when the Ledge Light station went offline.

of which are shown in Fig. 5.4. The distances between nodes N1 and N2, N1 and N3, and N2 and N3 were 1216, 840 and 556 meters, respectively.

Figure 5.4: Locations of the underwater nodes and weather stations.

5.3.2 Adversary model

I assume that Eve is an eavesdropper, i.e. passive adversary. She can hear all communications between Alice and Bob. At the same time, Eve knows the detailed structure of the probe, so she can measure the RSS between herself and Alice as well as Bob by overhearing probe signals. She can also extract the information such as the position of excursions in Mathur and the binary vector **E** in Patwari, since these messages have not been encoded by the secret key yet. Furthermore, I suppose that Eve understands how the key generation approach used by Alice and Bob works and knows parameters used in each approach exactly.

5.3.3 Performance metrics

The metrics I used to evaluate the performance of the three candidate approaches are listed as follows:

- *Key generation rate*: The average number of secret key in terms of bits extracted from each RSS measurement.
- *Bit mismatch rate*: The ratio of the mismatched bits between the keys produced by Alice and Bob to the length of secret key.
- *Leakage rate*: The ratio of the matched bits between keys produced by Eve and Alice to the length of secret key.
- *Randomness*: "A series of numbers is random if the smallest algorithm capable of specifying it to a computer has about the same number of bits of information as the series itself" [121].

Key generation rate and bit mismatch rate are the most important two metrics as they describe the capability of a key generation approach to produce valid bits from the raw RSS measurements, which in turn affects the minimum time needed to build a secret connection between Alice and Bob. In general, these two metrics characterize the efficiency and the correctness of a RSS based key generation approach.

Leakage rate measures that how many information Eve can guess from overhearing the communication between Alice and Bob. Assume Alice is the initiator of a key generation process, I call that one bit is leaked from Alice to Eve if both the position (bit ID) and the value of this bit are the same between the keys generated by Eve and Alice.

Randomness is a critical metric to measure the predictableness of secret bits. For a key with low randomness, once Eve gets the pattern of the key, she can predict the content of this key easily without further RSS processing. To prevent this problem, a key used by Alice and Bob should have high randomness to support long-term secret communications. I evaluate the randomness of a key based on the approximate entropy test, which is provided by NIST test suite [122]. The significance level I used in tests is 0.01, which implies that we would expect one out of one hundred sequences to be

rejected by a randomness test if the sequence is truly random. The tested key is accepted as a random sequence if and only if the P-value of its approximate entropy is larger than or equal to 0.01, otherwise it is rejected.

5.3.4 RSS correlation test

The performance of RSS based secret key generation depend highly on the symmetry of RSS sequences between two key generation parties, namely Alice and Bob. However, even if the acoustic channel is reciprocal, the symmetry of RSS measurements may be significantly affected by other factors, such as asynchronous transmission of probe signals between Alice and Bob. If the difference between the transmission time of one pair of probes (the probes from Alice and Bob with the same ID) is larger than the correlation time of RSS sequence, the measurement of RSS on two users could be significantly different. Therefore, it is important to study the correlation feature of RSS sequence to choose the probe transmission schemes, or evaluate the performance of a key generation approach.

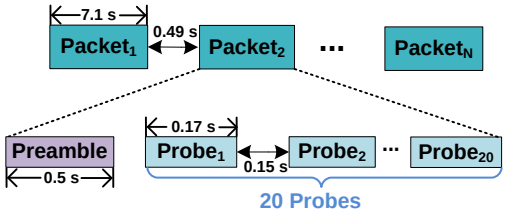


Figure 5.5: The data structure in RSS correlation estimation.

In order to estimate the correlation of RSS sequence, in the experiment I let node N2 to broadcast long packets successively. Each long packet had one preamble segment and 20 probes, the transmission time of which were 500 and 170 milliseconds respectively. The total transmission time of a long packet was about 7.1 seconds. The time intervals (guard time) between neighboring probes and adjacent packets were 150 and 490 milliseconds respectively, as shown in Fig. 5.5.

To eliminate the effect of lost probes, only the packets received by both N1 and N3 were used to estimate the correlation time of RSS sequence. Taking the reception on N3 as an example, assume it has fully received a total number of N_l long packets. Denote $x_{i,j}$ as the signal-to-noise ratio (SNR) measured by the probe j in the packet i, where $i = 1, 2, \ldots, N_l$ and $j = 1, 2, \ldots, 20$. Let vector $\mathbf{X}_i$ be equal to $[x_{i,1}, x_{i,2}, \ldots, x_{i,20}]^{\mathrm{T}}$, then we can estimate the correlation of the receiving SNRs of twenty neighboring RRS measurements by computing the covariance matrix of $\mathbf{X}_i$ according to

$$\mathbf{C}_{\mathbf{X}} = \frac{1}{N_l - 1} \sum_{i=1}^{N_l} (\mathbf{X}_i - \mu_i)(\mathbf{X}_i - \mu_i)^{\mathrm{T}}, \tag{5.1}$$

where $\mu_i = \frac{1}{N_l} \sum_{i}^{N} \mathbf{X}_i$, is the mean vector of $\mathbf{X}_i$.

5.3.5 Probe transmission

In the key generation tests, Alice was configured as the initiator for probe exchange. She used 20 watts to sent a single probe signal to Bob, who then reply a same probe as response. This probe exchange repeated periodically every w_t seconds. The sender ID, probe ID and stamp of local transmission time were attached in each probe.

With the above mentioned probe transmission strategy, the time difference between the transmissions of one pair of probes was the sum of propagation delay, transmission time of a probe signal and switching time of acoustic modem from receiving mode to sending mode, which are denoted as t_p, t_t and t_s respectively as shown in Fig. 5.6. The values of t_t and t_s vary with different acoustic modem, which were 500 and 200 milliseconds respectively in my sea tests. It is easy to calculate that the minimal time difference between transmissions of a pair of probes was about 1.07 seconds, when N2 and N3 played roles as Alice and Bob.

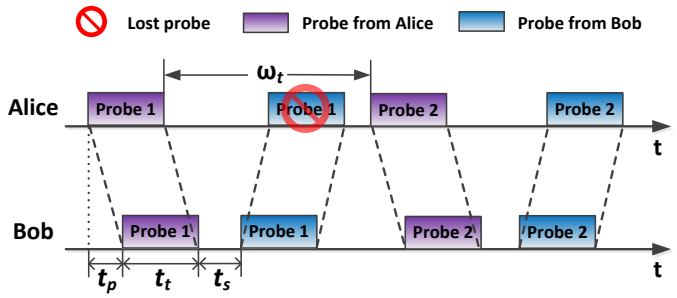
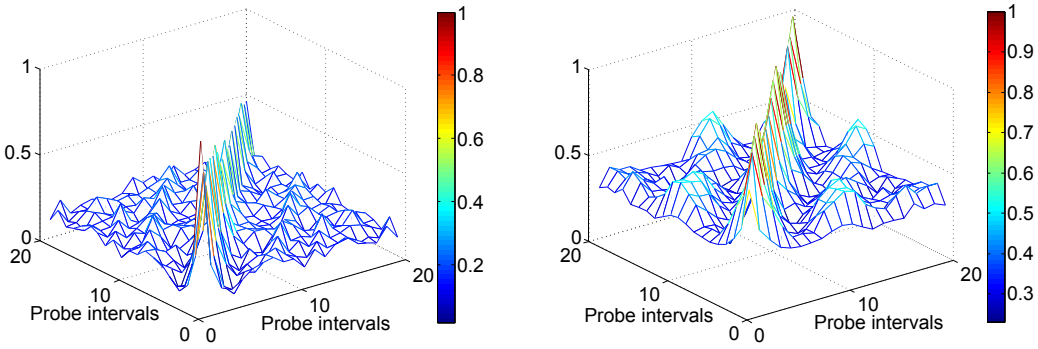


Figure 5.6: The time line of prob transmissions between Alice and Bob.

5.4 Experiment results and analysis

In this section, I present the sea test results conducted at Long Island Sound. I first introduce the correlation features of RSS sequences collected in the experiments, and then compare the average performance of the three candidate key generation approaches in various weather conditions.

5.4.1 RSS correlation



(a) Node 1 (b) Node N3

Figure 5.7: Normalized correlation coefficient as a function of intervals between two probes

As illustrated in Fig. 5.6, the long propagation delay in water introduced nonnegligible time differences between the transmissions of probes on two nodes. If this time

difference is larger than the correlation time of the acoustic channel, it fails the symmetrical assumption on the communications, which is the fundamental of RSS based secret key generation schemes. Therefore studying the correlation feature of the acoustic channel is a critical issue in real sea experiments.

Fig. 5.7 demonstrates the normalized correlation of SNR sequence[4] measured on nodes N1 and N3 in March 4, 2014. The average wind speed during the channel test was 4.4 knots. In the experiment, node N2 broadcasted 300 long packets, the structure of which is shown in Fig. 5.5. After removing the packets with lost probes, I finally collected 5860 measurement points from a total number of 293 packets to evaluate the correlation of SNR sequences received on both nodes N1 and N3.

From Fig. 5.7 I observe that the correlation coefficient of receiving SNR reduces quickly with the increased intervals between two probes on both receivers. However, the average correlation coefficient on N3 is much higher than that of on N1. More specifically, the normalized coefficient on N1 decreases from 1 to 0.5 within one probe interval, which is $(0.15 + 0.17) = 0.32$ seconds, but the time is extended to 3 probe intervals, which is 0.96 seconds on N3. If we consider the RSS sequence is correlated when the normalized correlation coefficient is higher than 0.5, the correlation time of RSS sequence on N3 is about 1 second, considerably longer than the 0.32 seconds on N1.

There are two factors affecting the RSS correlation time on N1 and N3. Firstly, the average receiving SNR on N1 was higher than that of N3, owing to a shorter distance to N2. The random noise, which reduces the correlation of SNR sequence, thus took less effects on N3 than that of on N1. Secondly, the dynamic of acoustic channel tends to grow with distance. Hence the correlation of receiving SNRs reduced faster at N1 than at N3. When node N2 initiate secret communication with N1 or N3, a longer

[4]Each measurement point is the average of reception SNR on four hydrophones of an acoustic modem.

correlation time of RSS sequence will allow N3 a larger allowance on the difference between the transmission time of one pair of probes, which, in turn, reduces the bit mismatch rate of the secret key.

5.4.2 Performance comparison

To evaluate performance of the three approaches, I conducted experiments where nodes N2, N3 and N1 played the roles of Alice, Bob and Eve, respectively. Table 5.3 lists the average performance comparison results, and the detailed analysis are as follows.

Table 5.3: Performance Comparison of Original Approaches

Approach	Key Generation Rate	Bit Mismatch Rate	Leakage Rate	Entropy	P-vaule
Aono	0.10	0.108	0.094	0.34	< 0.01
Mathur	0.09	0.385	0.141	0.68	0.43
Patwari	3.00	0.493	0.324	0.69	0.61

5.4.2.1 Key generation rate comparison

The key generation rates of Aono and Patwari only depend on the parameters l or k in Table 5.1. Limited by single-bit extraction, the maximum generation rate of Aono cannot exceed one. Patwari, instead, has much higher key generation rate with the multi-bit extraction strategy. Among three approaches listed in Table 5.3, Mathur has the lowest key generation rate, which is not only affected by the high dynamic feature of underwater acoustic channel, but also determined by the parameters m and α. Specifically, after large-scale fading elimination, the fluctuation among neighboring RSS measurements in Mathur is usually very large, and causes a low chance of getting an excursion of consecutive RSS measurements above or below the thresholds. As shown in Fig. 5.8, in Mathur a high key generation rate is excepted with small m and α ($m = 1$ and $\alpha = 0.05$), owing to a high probability to get a short excursion of

consecutive high or low RSS measurements. However, the key generation rate reduces significantly with the growth of m and α, as more RSS measurements will fall between two thresholds and cannot be used for a reliable key extraction.

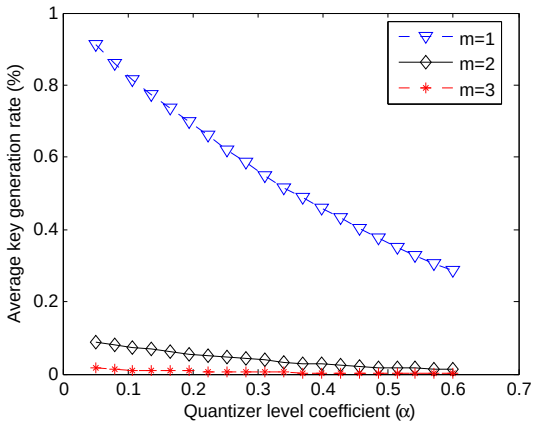


Figure 5.8: The average key generation rate of Mathur with respect to m and α.

5.4.2.2 Bit mismatch rate

Due to the imperfect reciprocity of acoustic channels and asynchronous probe transmissions between Alice and Bob, the bit mismatch rates were significant in my experiment. Mathur and Patwar have much higher bit mismatch rate than Aono, due to the fact that these two keys generations are susceptible to the small-scale variations of RSS measurements. Resulted from the random fluctuation of an acoustic channel, the small-scale variations have a low reciprocity. The keys in Aono, on the other hand, are produced from the "envelope" of a RSS sequence, which is dominated by the large-scale fading, and have a good symmetry feature in oceans as shown in Fig. 5.9. In addition, in order to maximize the key generation rate, Patwari tries to quantize every RSS measurement point into multiple secret bits without any censoring scheme. This further increases its bit mismatch rate, as the key extraction in Patwari is not able

to tolerate a large difference on the RSS sequences between Alice and Bob in a high dynamic underwater environment.

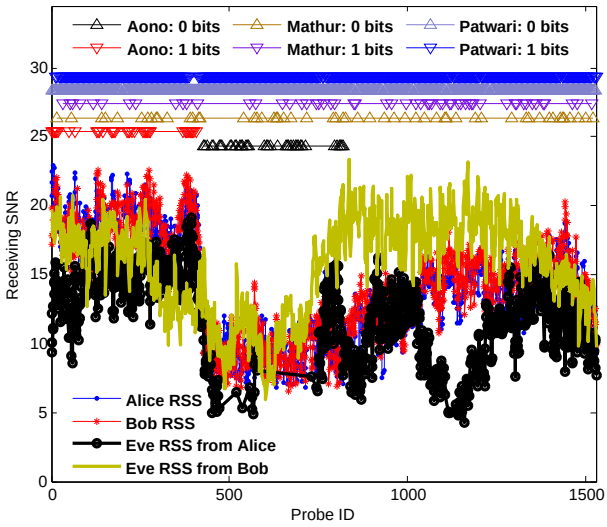


Figure 5.9: The raw RSS traces of Alice, Bob and Eve, and bits of the secret key generated by Alice with three different approaches.

The bit mismatch rate of Mathur can be reduced by using a large m or α. As shown in Fig. 5.10, when m increases from 1 to 3 with $\alpha = 0.5$, the corresponding bit mismatch rate can be reduced from 45% to less than 35%. If we increase α from 0.05 to 0.5 at the same time, the mismatch rate can be further reduced to about 11%, which is only 1/4 of the situation with $m = 1$ and $\alpha = 0.05$. However, the bit mismatch rate improvement is at the cost of decreasing in key generation rate, as shown in Fig. 5.8. Essentially, there is a tradeoff between key generation rate and bit mismatch rate in Mathur. Larger m and α improve the reliability of data picked up from raw RSS sequence for key generation, but with a penalty of removing highly fluctuate points, which may be potentially used for the key extraction.

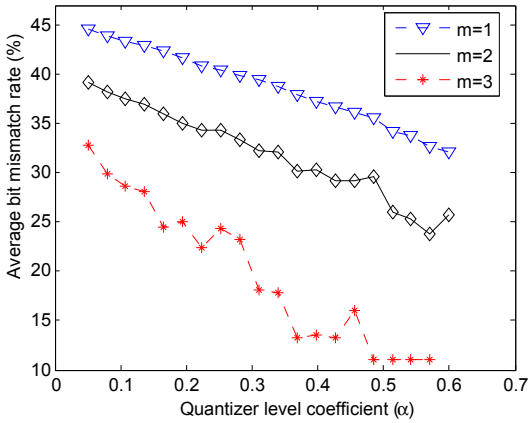


Figure 5.10: Average bit mismatch rates of Mathur with varied m and α.

5.4.2.3 Leakage rate

The leakage rate of Aono is lower than 10%, since the occurring time of a larger-scale fading in acoustic channel between Alice and Bob is very different from the one between Alice and Eve. As shown in Fig. 5.9, the corresponding prob IDs of a larger-scale fading in the channel from Alice to Bob are between 420 to 820, but the probes are between 1040 and 1220 for the channel from Alice to Eve[5] , which means that positions of bits "0" in the keys of Eve and Alice may be very different according to the key extraction method in Aono. Moreover, the leakage rate of Mathur in Table 5.3 is also low, which indicates that either the positions or the values of excursions in RSS sequences of Alice and Eve are dissimilar. In contrast, Patwari has a high leakage rate over 30%, which means 30% of bits are the same between the keys produced by Alice and Eve mainly due to the following two reasons:

a. Patwari is a multi-bit approach. Even through two RSS points are quantized into different intervals, a part of their secret bits may still be the same with each other. Using $k = 4$ as an example, due to the asymmetrical response of RSS sequence, Alice and Eve classify a RSS point into the 1^{st} and the 8^{th} intervals respectively,

[5]We cannot assert that the probes between 470 and 740 for the channel from Alice to Eve are larger-scale fading or not, since most of these probes from Alice are lost by Eve in the experiment.

which are represented by 00 00 and 10 00 after the 4-bit quantization. It is easy to observe that the last three bits produced by Alice and Eve are the same.

b. The Gray codes, which are used by Patwari to improve the bit mismatch rate, also "erase" some differences on quantized RSS sequences between Alice and Eve. One extreme example is that if we use an ideal code scheme, which can eliminate all differences on the keys between Alice and Bob, the bit mismatch rate reduces zero. However, the leakage rate will be 100%, since Eve can also perfectly correct her key to make it as same as the key used by Alice and Bob, no matter the channel is reciprocal or not.

5.4.2.4 Randomness

In my tests, the keys produced by Mathur and Patwari both have high P-values to pass the randomness test with a significance level of 0.01, but the ones produced by Aono does not pass the approximate entropy test. As shown in Fig. 5.9, due to the effect of large-scale fading, Aono's key usually consists of a large number of continuous ones and zeros, which significantly decreases its approximate entropy and randomness.

5.5 Performance improvement

As discussed earlier, the long transmission time of probe signal leads to low key generation rate. The randomness of the key mainly depends on the large-scale fading of an acoustic channel, while the asymmetric RSS measurements result in high key disagreements. In this section, I provide some feasible solutions to improve RSS based key generation in terms of key generation rate, randomness and key agreement probability.

5.5.1 Improvement on key generation rate

Intuitively, we can increase the key generation rate by using a multi-bit approach in the key extraction stage. However, compared with single-bit methods, the conventional multi-bit RSS-based key generation scheme is susceptible to the imperfect asymmetry between RSS measurements, thereby resulting in a high disagreement probability between the shared keys. As listed in Table 5.3, the bit mismatch rate of the multi-bit approach proposed by Patwari is near 50% in oceans, which requires an overhaul before applying it into UANs.

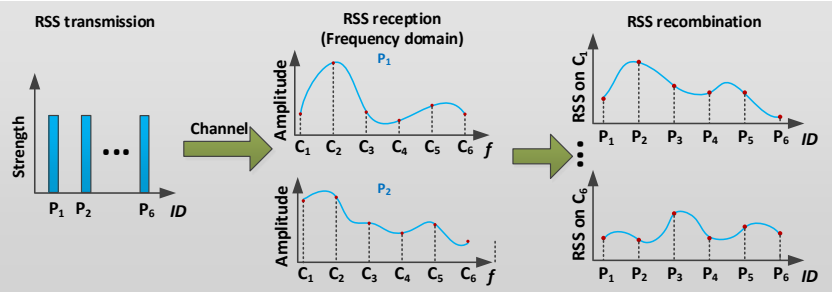


Figure 5.11: Multi-channel key generation scheme, where P_i is the probe ID and C_i is the ID of sub-channel.

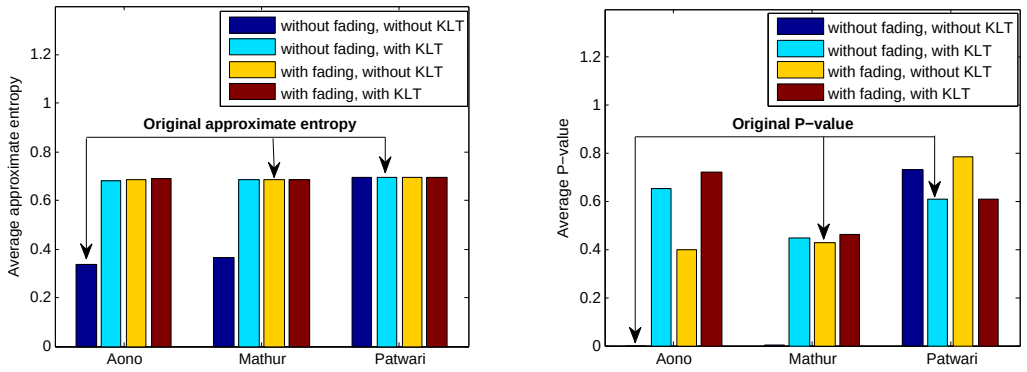
To keep the advantages of low bit-mismatch rate in single-bit methods and the high key generation rate in multi-bit approaches, a viable solution is to use the scheme of multi-channel key generation. In this scheme, the nodes divide the communication bandwidth into multiple independent sub-channels, and the probe signal, e.g., an orthogonal frequency-division multiplexing (OFDM) signal, should have at least one frequency component on each sub-channel. After receiving the probes, the receiver transforms the received signal to the frequency domain. With this scheme, the RSS measurements can be performed on each sub-channel producing independent RSS sequences. The communicating parties can harvest a total number of N_{sub} RSS measurements from each probe reception, where N_{sub} is the number of sub-channels, as shown in Fig. 5.11.

Obviously, the more sub-channels we use, the higher key generation rate we can achieve. However, there is a possibility that the RSS measured at neighboring sub-channels are correlated if their frequency interval is small. In this case, the randomness of the secret bits will decrease. For this reason, we should choose the sub-channels not close to each other in the frequency domain.

5.5.2 Improvement on key randomness

In order to increase the randomness of a key, Mathur and Patwari employ the signal preprocessing mechanisms, namely, the large-scale fading elimination and the KLT decorrelation transformation, respectively. One interesting question is can these two preprocessing approaches improve the key's randomness on other key generation methods.

Fig. 5.12 presents a comparison of approximate entropy and P-value with and without the large-scale fading elimination and the KLT decorrelation for the three approaches. I observe that either the fading elimination or the KLT decorrelation alone can effectively increase the entropy and the P-value ofs keys produced by Aono and Mathur. Nevertheless, we cannot further improve their randomness by using both signal preprocessing mechanisms at the same time. On the contrary, no matter with or without the fading elimination and the KLT decorrelation, the key produced by Patwari can always keep a high randomness. This is because Patwari quantizes each measurement point into multiple bits. Therefore, if the correlation between neighboring measurements is low, the secret bits used to represent each RSS point are also very different. In other words, the high dynamic feature of RSS sequence in oceans has already allowed the key generated by Patwari to have a good randomness, especially in the situation with a large number of bits for each quantization (a large k).



(a) Average approximate entropy

(b) Average P-value

Figure 5.12: The approximate entropy and P-value with and without large-scale fading elimination and KLT decorrelation

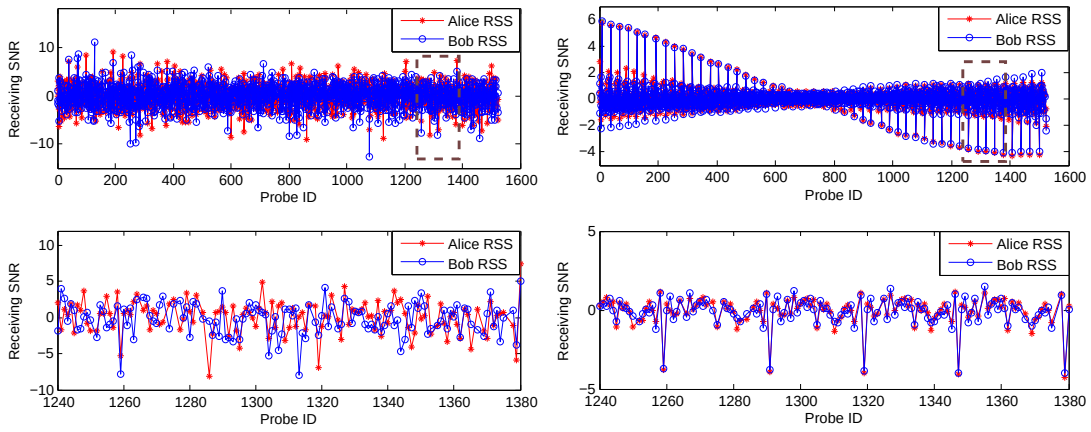
5.5.3 Improvement on key agreement probability

Evidentially, the key disagreement in RSS-based key generation approaches can be reduced by improving the symmetry of RSS sequences between two communicating entities. However, due to the large time difference of non-concurrent RSS transmissions in oceans, there could exist considerable discrepancies on RSS measurements. Therefore, the interpolation filter, which is usually adopted in radio networks, may fail in UANs. Here I advocate to use the smooth filter in the signal preprocessing stage to improve the key robustness.

The smooth filter has been widely applied in many areas, such as statistics and image precessing. It is an efficient way to capture the critical features in a data, while removing the fast varying components, like the noise and interference. By using a smooth filter in the RSS-based key generation approach, the two parties can reduce the random fluctuations in their RSS measurements and thus decrease the bit mismatch rate of the key.

According to the environmental conditions, we can select different smooth filters to achieve a good performance for the key generation. For instance, if the probe signals are polluted by strong ambient noise, a Savitzky-Golay filter or a symmetric moving

average filter is recommended to improve the reciprocity of RSS sequences. If the symmetry of RSS sequences is degraded by a burst interference, such as the signal from a sonar or a marine mammal, the robust local polynomial regression (LOESS) filter is preferred in this situation.



(a) Without the smooth filter ($S_w = 1$) (b) With the smooth filter ($S_w = 39$)

Figure 5.13: The RSS sequences with and without the moving average low-pass smooth filter. (Used the same raw data with Fig. 5.9)

I compare the receiving SNR sequences between Alice and Bob with and without the moving average low-pass smooth filter in Fig. 5.13. Without the smooth filtering, there are intensive fluctuations in the RSS measurements both on Alice and Bob. Using the Alice's RSS sequence as an example, the standard deviation of the sequence in Fig. 5.13 (a) is as high as 2.35 decibels. At the same time, there is a large difference between RSS values measured by Alice and Bob with a same probe ID, which is 2.20 decibels in average. In contrast, after going through a moving average smooth filter with window size $S_w = 39$ in Fig. 5.13 (b), the standard deviation of the RSS sequence on Alice decreases to 0.81 decibels, indicating a decreased jitter on RSS sequence. In addition, the average of RSS difference between Alice and Bob is dramatically reduced from 2.20 to 0.27 decibels. This means that the symmetry of RSS sequence is significantly

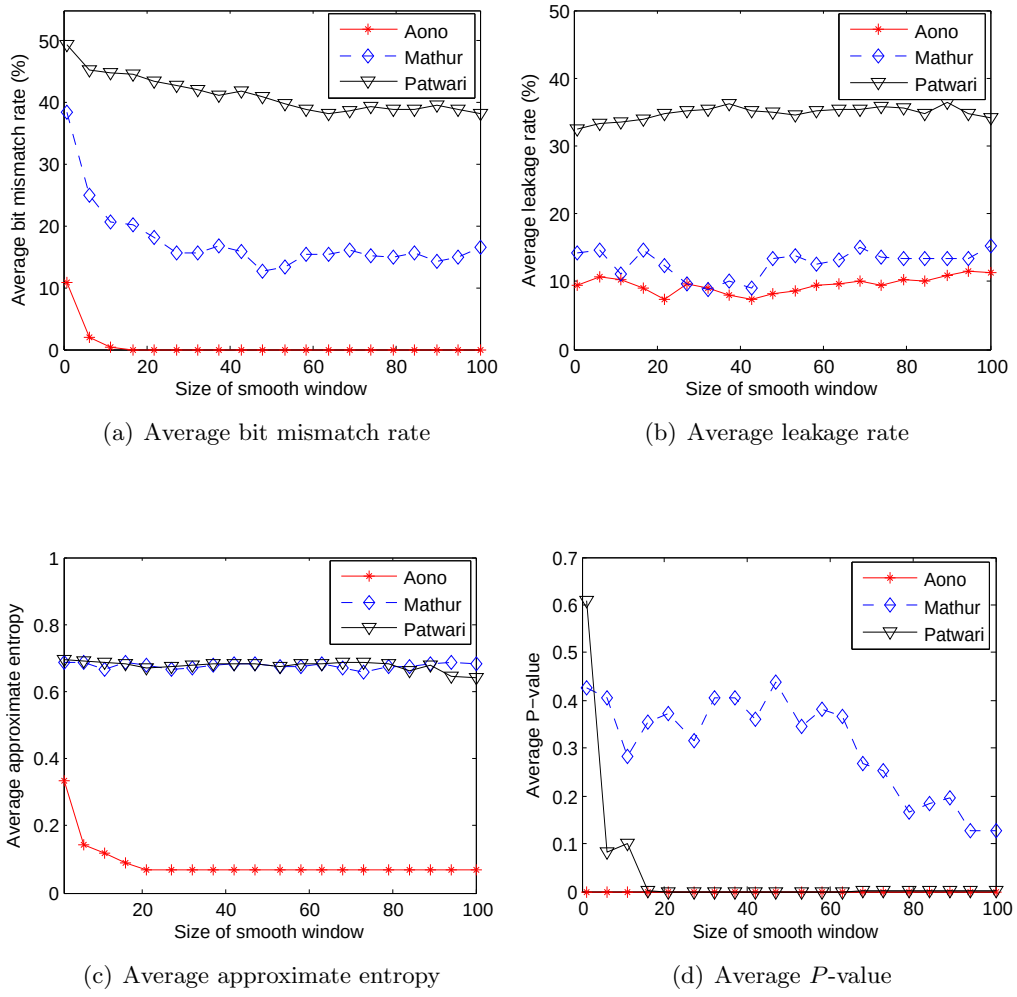
improved with the smooth filter, which can also be easily observed in the detailed view of RSS response between probe 1240 to probe 1380 in Fig. 5.13.

I evaluate the performance of three key generation schemes in terms of bit mismatch rate, leakage rate, approximate entropy and P-value in Fig. 5.14 with respect to the size of smooth window, S_w. The results verify the efficiency of a smooth filter on reducing the bit mismatch rates for the three representative RSS-based key generation approaches. As demonstrated in Fig.5.14, the bit mismatch rates of all three approaches reduce with the increased size of smooth window significantly. More specifically, by using the symmetric moving average smooth filter in the experiments, the average bit mismatch rates of Aono, Mathur and Patwari dramatically decrease by 100%, 63% and 20%, respectively.

It is worth noting that, there is a tradeoff between the mismatch rate and the randomness of the secret key when a smooth filter is applied. According to the experiment results, the P-value of the approximate entropy test would be less than 0.01 for Mathur's and Patwari's key generation approaches if the size of the smooth window is over 15. The detailed discussions are as follows.

a. As shown in Fig. 5.14 (a), the average bit mismatch rates of Aono and Mathur have significant reduction when smooth filter is used. However this improvement is less effective in Patwari, which finally loiters around 40% in large S_w conditions. This is because Patwari, as a multi-bit approach, is more sensitive to the asymmetry of RSS responses than single-bit methods, especially in a situation with high numbers of bit for each quantization (large k).

b. When applying the smooth filter in each key generation approach, the information leakage from Alice to Eve has no obvious increase, as shown in Fig. 5.14 (b). This indicates that the smooth filter does not "erase" the difference on RSS sequences between Alice and Eve, which is helpful to maintain the security of a key.



(a) Average bit mismatch rate

(b) Average leakage rate

(c) Average approximate entropy

(d) Average P-value

Figure 5.14: Performance comparison with respect to the size of smooth window. Parameter settings were described in Table 5.1.

c. By comparing Fig. 5.14 (c) and Fig. 5.14 (d) I observe that, with a larger smooth window, the approximate entropies of Patwari and Mathur are kept around 0.7, but the corresponding P-values of Patwari drop to zero quickly. This phenomenon arises in the repeated bit patterns with smooth filter. Taking the data I collected in November 24, 2013 as an example, after smoothing, the RSS values from probe 3 to probe 7 are all classified into the interval 32 in Patwari. The bit pattern 1 0 0 thus will be repeated five times after the multi-bit extraction. Compared with the key of Aono, in which a large mount of "1" or "0" stick together, the bits "1" of Patwari in this example increases the irregularity of the sequence, but the randomness of a key is still restrained by these repeated patterns.

d. The key generation rates of Aono and Patwari are predetermined by the number of successfully received probes and the corresponding parameters described in Table 5.1, which are constants in each test. Intuitively, smooth filter decreases the fluctuation between neighboring RSS points, which improves the formation of excursions in Mathur. However, I do not observe obvious changes on the key generation rate with a larger smooth window. This indicates that the smooth filter may not be helpful to increase the number of excursions in Mathur.

To summarize, the moving average smooth filter does not have significant effect on the leakage rate and the key generation rate of the three approaches. By adjusting the size of smooth window, there is a trade-off between the bit mismatch rate and the randomness of the key produced by each approach. According to the experiment results I have that the smooth filter with S_w around 9 is a decent one to guarantee each approach to get a good improvement on bit mismatch rate, while does not sacrifice too much on randomness.

5.6 Summary

In this chapter, I evaluated the RSS-based key generation approaches in underwater environments for secret acoustic communications. While these approaches have been well studied in terrestrial networks, they are facing many new challenges yet to be addressed due to the unique features of acoustic systems.

From the experiment results I observed that (a) the transmission time of a probe signal in UANs is much longer than that in radio networks and thus results in a low key generation rate, (b) due to the long propagation delay and large transmission time, the asymmetry of RSS measurements between two communicating parties is more significant in UANs than that in radio networks, which causes a high bit mismatch rate on the shared key, and (c) the randomness of some RSS key generation approaches is highly sensitive to the large-scale fading of acoustic channel, which may cause a large number of continuous "1" or "0" in the key.

Finally, I introduced three solutions to improve the performance of RSS-based key generation approaches in terms of key generation rate, randomness and bit mismatch rate. The multi-channel key generation scheme enables communicating parties to extract secret bits on multiple sub-channels, therefore improves the efficiency of key generation significantly. The KLT decortication technique can help to eliminate the large-scale fading in the acoustic channel and improve the entropy and the P-value of key produced through RSS measurement. The smooth filter can improve the symmetry of RSS measurements, thereby reducing the discrepancies between the shared keys. According to experiment results from sea trails I have that by using the symmetric moving average smooth filter, the average bit mismatch rates of the approaches proposed by Aono, Mathur and Patwari were decreased by 100%, 63% and 20%, respectively.

Chapter 6

Conclusions

In this dissertation, driven by practical issues, I explored the real system features of underwater acoustic systems and studied their implications on underwater MAC, synchronization and secure communication.

Firstly, I conducted real sea experiments and revealed the new discoveries on real system features, including the long preamble and long transmission delay, heterogeneous packet delivery, temporal and spatial transmission range uncertainty, multi-hop interference and delayed packet transmission. Discussions were provided based on the new discoveries, in hopes of giving some meaningful insights into the practical MAC and time synchronization design for real underwater networks. I have also implemented the real system features into the Aqua-Sim simulator, aiming to reduce the gap between the simulation and reality. The practical features of underwater systems will be available in the next release of Aqua-Sim.

Secondly, I analyzed and evaluated the random access based UW-Aloha, handshaking based SASHA and scheduling based PMAC in sea tests with different network scenarios. Based on the field test results, I study the advantages, shortcomings and limitations of three different MAC mechanisms and how they work in real systems.

Following this, I further explored the impact of real system features on general underwater MAC protocols and proposed a traffic estimation based receiver initiated MAC, called FERI MAC. In FERI MAC, receivers replace the role of senders in conventional MAC protocols as the initializer of a handshake process. It allows receiver to establish handshake with multiple senders in parallel, improving the network performance in terms of throughput, delivery delay and energy efficiency.

Thirdly, I conducted lab tests and measured the message delivery delays encountered in real acoustic system. Through experiment results, I revealed different sources of delays in UANs than that in TWNs. The delays include command detection time, command delivery time, command transmission time, transmission preparation time, transmission time, propagation time and reception time, among which the command detection time and transmission preparation time are significant and have the highest uncertainties. In addition, I evaluated representative time synchronization schemes, namely TSHL-RS and RBS-UW, in a lab environment and provided inspirations on practical time synchronization protocol design and performance improvement.

Fourthly, I explored RSS based key generation methods in UANs. More specifically, I discussed the advantages of RSS key generation and analyzed the grand challenges from the unique features of UANs. Furthermore, I conducted sea tests and evaluated the performance of three representative RSS based key generation approaches, namely Ano, Mathur and Patwari, for underwater secure communications. From sea experiment, I explored how underwater noise, asynchronous transmissions of probe signal and large-scale fading of acoustic channel affects the representative RSS based key generation protocols. Meanwhile, I provided solutions to improve the performance in terms of key generation rate, randomness and key agreement probability.

Bibliography

[1] Jun-Hong Cui, Jiejun Kong, Mario Gerla, and Shengli Zhou. The challenges of building mobile underwater wireless networks for aquatic applications. *IEEE Network*, 20:12–18, May 2006.

[2] Ian F. Akyildiz, Dario Pompili, and Tommaso Melodia. State-of-the-art in protocol research for underwater acoustic sensor networks. In *Proceedings of ACM WUWNet*, pages 7–16, September 2006.

[3] Mandar Chitre, Shiraz Shahabudeen, and Milica Stojanovic. Underwater Acoustic Communications and Networking: Recent Advances and Future Challenges. *Marine Technology Society*, 42(1):103–116, 2008.

[4] Mandar Chitre, Shiraz Shahabudeen, and Milica Stojanovic. Underwater acoustic communications and networking: Recent advances and future challenges. *Marine technology society journal*, 42(1):103–116, 2008.

[5] John Heidemann, Milica Stojanovic, and Michele Zorzi. Underwater sensor networks: applications, advances and challenges. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 370:158–175, 2012.

[6] Yu Luo, Lina Pu, Michael Zuba, Zheng Peng, and Jun-Hong Cui. Cognitive Acoustics: Making Underwater Communications Environment-friendly. In *Proceedings of ACM WUWNet*, pages 48–49. ACM, 2014.

[7] Yu Luo, Lina Pu, Michael Zuba, Zheng Peng, and Jun-Hong Cui. Challenges and Opportunities of Underwater Cognitive Acoustic Networks. *IEEE Transactions on Emerging Topics in Computing*, 2(2):198–211, 2014.

[8] Peng Xie, Zhong Zhou, Zheng Peng, Jun-Hong Cui, and Zhijie Shi. Sdrt: a reliable data transport protocol for underwater sensor networks. *Ad Hoc Networks*, 8(7):708–722, 2010.

[9] Shaobin Cai, Zhenguo Gao, DeSen Yang, and Nianmin Yao. A network coding based protocol for reliable data transfer in underwater acoustic sensor. *Ad Hoc Networks*, 11(5):1603–1609, 2013.

[10] Dario Pompili, Tommaso Melodia, and Ian F. Akyildiz. Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks. In *Proceedings of International Conference on Mobile Computing and Networking (MobiCom)*, pages 298–309, September 2006.

[11] Peng Xie, Li Lao, and Jun-Hong Cui. VBF: vector-based forwarding protocol for underwater sensor networks. In *Proceedings of IFIP Networking*, pages 1216–1221, May 2006.

[12] Josep Miquel Jornet, Milica Stojanovic, and Michele Zorzi. Focused beam routing protocol for underwater acoustic networks. In *Proceedings of ACM WUWNet*, pages 75–81, September 2008.

[13] Hai Yan, Zhijie Shi, and Jun-Hong Cui. DBR: Depth-based routing for underwater sensor networks. In *Proceedings of IFIP Networking*, pages 1–13, May 2008.

[14] Uichin Lee, Paul Wang, Youngtae Noh, Luiz FM Vieira, Mario Gerla, and Jun-Hong Cui. Pressure routing for underwater sensor networks. In *Proceedings of IEEE INFOCOM*, pages 1–9, 2010.

[15] Zheng Peng, Zhong Zhou, Jun-Hong Cui, and Zhijie Jerry Shi. Aqua-Net: an underwater sensor network architecture: design, implementation, and initial testing. In *Proceedings of IEEE OCEANS*, Biloxi, MS, October 2009.

[16] Nitthita Chirdchoo, Wee-Seng Soh, and Kee Chaing Chua. Aloha-based MAC Protocols with collision avoidance for underwater acoustic networks. In *Proceedings of IEEE INFOCOM*, pages 2271 – 2275, April 2007.

[17] A. Syed, Wei Ye, and John Heidemann. T-Lohi: a new class of MAC protocol for underwater acoustic sensor networks. In *Proceedings of IEEE INFOCOM*, 2008.

[18] Marcal Molins and Milica Stojanovic. Slotted FAMA: a MAC protocol for underwater acoustic networks. In *Proceedings of IEEE OCEANS*, Singapore, May 2006.

[19] Borja Peleato and Milica Stojanovic. Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks. *IEEE Communications Letters*, 11:1025–1027, 2007.

[20] M.J. Xiaoxing Guo; Frater, M.R.; Ryan. Design of a propagation-delay-tolerant mac protocol for underwater acoustic sensor networks. *IEEE Journal of Oceanic Engineering*, pages 170–180, 4 2009.

[21] Peng Xie and Jun-Hong Cui. R-MAC: an energy-efficient MAC protocol for underwater sensor networks. In *Proceedings of WASA*. Springer, 2007.

[22] Zheng Peng, Yibo Zhu, Zhong Zhou, Zheng Guo, and Jun-Hong Cui. COPE-MAC: a contention-based medium access control protocol with parallel reservation for underwater acoustic networks. In *Proceedings of IEEE OCEANS*, Sydney, Australia, May 2010.

[23] Y. Noh, P. Wang, Uichin Lee, D. Torres, and M. Gerla. DOTS: a propagation delay-aware opportunistic MAC protocol for underwater sensor networks. In *Proceedings of IEEE ICNP*, 2010.

[24] Haining Mo, Lina Pu, Yibo Zhu, Zheng Peng, Zaihan Jiang, and Jun-Hong Cui. Evaluating selective ARQ and slotted handshake based access in real world underwater networks. In *Proceedings of WASA*, pages 206–220. Springer, 2013.

[25] M. K. Park and V. Rodoplu. UWAN-MAC: an energy-efficient MAC protocol for underwater acoustic wireless sensor networks. *IEEE Journal of Oceanic Engineering*, 32(3):710–720, July 2007.

[26] Roee Diamant and Lutz Lampe. A hybrid spatial reuse MAC protocol for ad-hoc underwater acoustic communication networks. In *Proceedings of IEEE ICC*, pages 1–5, 2010.

[27] Chih-Cheng Hsu, Kuang-Fu Lai, Cheng-Fu Chou, , and Kate Ching-Ju Lin. ST-MAC: spatial-temporal MAC scheduling for underwater sensor networks. In *Proceedings of IEEE INFOCOM*, April 2009.

[28] Yu Luo, Lina Pu, Zheng Peng, Zhong Zhou, and Jun-Hong Cui. An efficient MAC protocol for underwater multi-user uplink communication networks. *Ad Hoc Networks*, 2015.

[29] Yu Luo, Lina Pu, Zheng Peng, Zhong Zhou, and Jun-Hong Cui. CT-MAC: a MAC protocol for underwater MIMO based network uplink communications. In *Proceedings of ACM WUWNet*, November 2012.

[30] Jun Liu, Zhaohui Wang, Michael Zuba, Zheng Peng, Jun-Hong Cui, and Shengli Zhou. JSL: joint time synchronization and localization design with stratification compensation in mobile underwater sensor networks. In *Proceedings of International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 2012.

[31] Jun Liu, Zhong Zhou, Zheng Peng, Jun-Hong Cui, Michael Zuba, and Lance Fiondella. Mobi-Sync: efficient time synchronization for mobile underwater sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 24(2):406–416, 2013.

[32] Son Le, Jun Liu, Zheng Peng, Jiaxing Che, Yu Luo, and Jun-Hong Cui. Seamark-assisted inertial navigation for autonomous underwater vehicles. In *Proceedings of ACM WUWNet*, pages 1–8, November 2014.

[33] Yibo Zhu, Xiaoyan Lu, Lina Pu, Yishan Su, Robert Martin, Micheal Zuba, Zheng Peng, and Jun-Hong Cui. Aqua-Sim: an NS-2 based simulator for underwater sensor networks. In *Proceedings of ACM WUWNet*, December 2013.

[34] Peng Xie, Zhong Zhou, Zheng Peng, Hai Yan, Tiansi Hu, Jun-Hong Cui, Zhijie Shi, Yunsi Fei, and Shengli Zhou. Aqua-Sim: an NS-2 based simulator for underwater sensor networks. In *Proceedings of IEEE OCEANS*, Biloxi, MS, October 2009.

[35] Chiara Petrioli and Roberto Petroccia. SUNSET: simulation, emulation and real-life testing of underwater wireless sensor networks. In *Proceedings of IEEE UComms*, 2012.

[36] Riccardo Masiero, Saiful Azad, Federico Favaro, Matteo Petrani, Giovanni Toso, Federico Guerra, Paolo Casari, and Michele Zorzi. DESERT underwater: an NS-Miracle-based framework to design, simulate, emulate and realize test-beds for underwater network protocols. In *Proceedings of IEEE OCEANS*, pages 1–10, Yeosu, Korea, 2012.

[37] Zheng Peng, Son Le, Michael Zuba, Haining Mo, Yibo Zhu, Lina Pu, Jun Liu, and Jun-Hong Cui. Aqua-TUNE: a testbed for underwater networks. In *Proceedings of IEEE OCEANS*, Santander, Spain, June 2011.

[38] Yibo Zhu, Lina Pu, Zigeng Wang, Xiaoyan Lu, Rashad Martin, Yu Luo, Zheng Peng, and Jun-Hong Cui. Underwater acoustic network protocol stacks: Simulator-based vs. os-based. In *Proceedings of IEEE OCEANS*, pages 1–7, St. John's, 2014.

[39] Y. Su, Y. Zhang, S. Le, H. Mo, L. Wei, Y. Huang, Z. Peng, and J. Cui. A versatile lab testbed for underwater sensor networks. In *Proceedings of IEEE OCEANS*, volume 1, September 2013.

[40] Milica Stojanovic, Josko A Catipovic, and John G Proakis. Phase-coherent digital communications for underwater acoustic channels. *IEEE Journal of Oceanic Engineering*, 19(1):100–111, 1994.

[41] Daniel Rouseff, Darrell R Jackson, Warren LJ Fox, Christopher D Jones, James A Ritcey, and David R Dowling. Underwater acoustic communication by passive-phase conjugation: Theory and experimental results. *IEEE Journal of Oceanic*

Engineering, 26(4):821–831, 2001.

[42] Baosheng Li, Shengli Zhou, Milica Stojanovic, Lee Freitag, and Peter Willett. Multicarrier communication over underwater acoustic channels with nonuniform doppler shifts. *IEEE Journal of Oceanic Engineering*, 33(2):198–209, 2008.

[43] Lina Pu, Yu Luo, Yibo Zhu, Zheng Peng, Shruti Khare, Jun-Hong Cui, Benyuan Liu, and Lei Wang. Impact of real modem characteristics on practical underwater MAC design. In *Proceedings of IEEE OCEANS*, Yeosu, Korea, May 2012.

[44] Chiara Petrioli, Roberto Petroccia, and John Potter. Performance evaluation of underwater MAC protocols: from simulation to at-sea testing. In *Proceedings of IEEE OCEANS*, Santander, Spain, June 2011.

[45] Lina Pu, Yu Luo, Haining Mo, Son Le, Zheng Peng, Jun-Hong Cui, and Zaihan Jiang. Comparing underwater mac protocols in real sea experiments. *Computer Communications*, 56:47–59, 2015.

[46] Lina Pu, Yu Luo, Haining Mo, Zheng Peng, Jun-Hong Cui, and Zaihan Jiang. Comparing underwater mac protocols in real sea experiment. In *Proceedings of IFIP Networking Conference*, pages 1–9, 2013.

[47] Son Le, Yibo Zhu, Zheng Peng, Jun-Hong Cui, and Zaihan Jiang. PMAC: a real-world case study of underwater MAC. In *Proceedings of ACM WUWNet*, December 2013.

[48] Affan Syed, Wei Ye, Bhaskar Krishnamachari, and John Heidemann. Understanding spatio-temporal uncertainty in medium access with ALOHA protocols. In *Proceedings of ACM WUWNet*, 2007.

[49] Chih-Cheng Hsu, Kuang-Fu Lai, Cheng-Fu Chou, and Kate Ching-Ju Lin. ST-MAC: Spatial-temporal mac scheduling for underwater sensor networks. In *Proceedings of IEEE INFOCOM*, pages 1827–1835, 2009.

[50] J. Heidemann, W. Ye, J. Wills, A. Syed, and Y. Li. Research challenges and applications for underwater sensor networking. In *Proceedings of Wireless Communications and Networking Conference (WCNC)*, pages 228–235, April 2006.

[51] Kenneth D Frampton. Acoustic self-localization in a distributed sensor network. *IEEE Sensors*, 6(1):166–172, 2006.

[52] Zhong Zhou, Shengli Zhou, Shuguang Cui, and Jun-Hong Cui. Energy-efficient cooperative communication in clustered wireless sensor networks. *IEEE Transactions on Vehicular Technology*, 57(6):3618–3628, 11 2008.

[53] Yu Luo, Lina Pu, Zheng Peng, Zhong Zhou, Jun-Hong Cui, and Zhaoyang Zhang. Effective relay selection for underwater cooperative acoustic networks. In *Proceedings of International Conference on Mobile Ad-Hoc and Sensor Systems (MASS)*, pages 104–112. IEEE, 2013.

[54] David L Mills. Internet time synchronization: the network time protocol. *IEEE Transactions on Communications*, 39(10):1482–1493, 1991.

[55] Fikret Sivrikaya and Bülent Yener. Time synchronization in sensor networks: a survey. *IEEE Network*, 18(4):45–50, 2004.

[56] Saurabh Ganeriwal, Ram Kumar, and Mani B Srivastava. Timing-sync protocol for sensor networks. In *Proceedings of International Conference on Embedded Networked Sensor Systems (Sensys)*, pages 138–149. ACM, 2003.

[57] Miklós Maróti, Branislav Kusy, Gyula Simon, and Ákos Lédeczi. The flooding time synchronization protocol. In *Proceedings of International Conference on Embedded Networked Sensor Systems (Sensys)*, pages 39–49. ACM, 2004.

[58] Jeremy Elson, Lewis Girod, and Deborah Estrin. Fine-grained network time synchronization using reference broadcasts. In *Proceedings of Symposium on Operating Systems Design and Implementation*, pages 147–163, 2002.

[59] Mihail L Sichitiu and Chanchai Veerarittiphan. Simple, accurate time synchronization for wireless sensor networks. In *Proceedings of Wireless Communications and Networking (WCNC)*, volume 2, pages 1266–1273. IEEE, 2003.

[60] Affan A Syed and John S Heidemann. Time synchronization for high latency acoustic networks. In *Proceedings of INFOCOM*, 2006.

[61] Nitthita Chirdchoo, Wee-Seng Soh, and Kee Chaing Chua. MU-Sync: a time synchronization protocol for underwater mobile networks. In *Proceedings of ACM WUWNet*, pages 35–42, 2008.

[62] Weichao Wang, Jiejun Kong, Bharat Bhargava, and Mario Gerla. Visualisation of wormholes in underwater sensor networks: a distributed approach. *International Journal of Security and Networks*, 3(1):10–23, 2008.

[63] Jiejun Kong, Zhengrong Ji, Weichao Wang, Mario Gerla, Rajive Bagrodia, and Bharat Bhargava. Low-cost attacks against packet delivery, localization and time synchronization services in under-water sensor networks. In *Proceedings of ACM Workshop on Wireless Security*, pages 87–96, 2005.

[64] Michael Zuba, Zhijie Shi, Zheng Peng, and Jun-Hong Cui. Launching denial-of-service jamming attacks in underwater sensor networks. In *Proceedings of ACM WUWNet*, page 12, 2011.

[65] Yi Huang, Peng Xiao, Shengli Zhou, and Zhijie Shi. A half-duplex self-protection jamming approach for improving secrecy of block transmissions in underwater acoustic channels. *IEEE Sensors Journal*, Jun. 2015.

[66] Adrian Perrig, Robert Szewczyk, Justin Douglas Tygar, Victor Wen, and David E Culler. SPINS: security protocols for sensor networks. *Wireless networks*, 8(5):521–534, 2002.

[67] Arvinderpas S Wander, Nils Gura, Hans Eberle, Vipul Gupta, and Sheueling Chang Shantz. Energy analysis of public-key cryptography for wireless sensor networks. In *Proceedings of Pervasive Computing and Communications (PerCom)*, pages 324–328. IEEE, 2005.

[68] Haowen Chan, Adrian Perrig, and Dawn Song. Random key predistribution schemes for sensor networks. In *Proceedings of Symposium on Security and Privacy*, pages 197–213. IEEE, 2003.

[69] Wenliang Du, Jing Deng, Yunghsiang S Han, Pramod K Varshney, Jonathan Katz, and Aram Khalili. A pairwise key predistribution scheme for wireless sensor networks. *ACM Transactions on Information and System Security (TISSEC)*, 8(2):228–258, 2005.

[70] Tomoyuki Aono, Keisuke Higuchi, Takashi Ohira, Bokuji Komiyama, and Hideichi Sasaoka. Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels. *IEEE Transactions on Antennas and Propagation*, 53(11):3776–3784, 2005.

[71] Suhas Mathur, Wade Trappe, Narayan Mandayam, Chunxuan Ye, and Alex Reznik. Radio-telepathy: extracting a secret key from an unauthenticated wireless channel. In *Proceedings of International Conference on Mobile Computing*

and Networking (MobiCom), pages 128–139. ACM, 2008.

[72] Neal Patwari, Jessica Croft, Suman Jana, and Sneha Kumar Kasera. High-rate uncorrelated bit extraction for shared secret key generation from channel measurements. *IEEE Transactions on Mobile Computing*, 9(1):17–30, 2010.

[73] Robert Wilson, David Tse, and Robert A Scholtz. Channel identification: Secret sharing using reciprocity in ultrawideband channels. *IEEE Transactions on Information Forensics and Security*, 2(3):364–375, 2007.

[74] Maxime Guillaud, Dirk TM Slock, and Raymond Knopp. A practical method for wireless channel reciprocity exploitation through relative calibration. In *Proceedings of International Symposium on Signal Processing and Its Applications*, pages 403–406, 2005.

[75] Yi Huang, Shengli Zhou, Zhijie Shi, and Lifeng Lai. Experimental study of secret key generation in underwater acoustic channels. In *Proceedings of Asilomar Conference on Signals, Systems and Computers*, November 2014.

[76] Gregory D Durgin. *Space-time wireless channels*. Prentice Hall Professional, 2003.

[77] Lina Pu, Yu Luo, and Jun-Hong Cui. Exploring underwater synchronization protocols in real systems. Technical Report Technical Report: UbiNet-TR15-01, UCONN CSE, 2014.

[78] Yu Luo, Lina Pu, Zheng Peng, and Zhijie Shi. RSS-based secret key generation in underwater acoustic networks: advantages, challenges and performance improvements. *IEEE Communications Magazine*, February 2016. (Accepted).

[79] Milica Stojanovic. On the relationship between capacity and distance in an underwater acoustic communication channel. In *Proceedings of ACM WUWNet*, September 2006.

[80] V. Capellano. Performance improvement of a 50km acoustic transmission through adaptive equalizationi and spatial diversity. In *Proceedings of IEEE OCEANS*, Nova Scotia, Canada, October 1997.

[81] Zheng Peng, Son Le, Michael Zuba, Haining Mo, Hao Zhou, Jun-Hong Cui, Shengli Zhou, Zaihan Jiang, and Jeffrey A. Schindall. Field test experience of an underwater wireless network in the atlantic ocean. In *Proceedings IEEE OCEANS*, Bergen, Norway, June 2013.

[82] Teledyne Benthos Incorporation. Benthos modem. teledynebenthos.com [Online]. Available: `http://teledynebenthos.com/product`. [Accessed: May, 2015].

[83] LinkQuest Incorporation. Underwater acoustic modem models. link-quest.com [Online]. Available: `http://www.link-quest.com/html/models1.htm`. [Accessed: May, 2015].

[84] Evologics GmbH Incorporation. Evologics modem. evologics.de [Online]. Available: `http://www.evologics.de/en/products/acoustics/index.html`. [Accessed: May, 2015].

[85] L. Freitag, M. Grund, S. Singh, J. Partan, P. Koski, and K. Ball. The WHOI Micro-Modem: an acoustic communications and navigation system for multiple platforms. In *Proceedings IEEE OCEANS*, Washington, DC, September 2005.

[86] Zheng Peng, Haining Mo, Jun Liu, Zhaohui Wang, Xiaoka Xu, Son Le, Yibo Zhu, Jun-Hong Cui, Zhijie Shi, and Shengli Zhou. NAMS: a networked acoustic

modem system for underwater applications. In *Proceedings of IEEE OCEANS*, Kona, Hawaii, September 2011.

[87] M.M. Wang, A. Agrawal, A. Khandekar, and S. Aedudodla. Preamble design, system acquisition, and determination in modern ofdma cellular communications: an overview. *IEEE Communications Magazine*, 49(7):164–175, 2011.

[88] Lina Pu, Yu Luo, Zheng Peng, Haining Mo, and Jun-Hong Cui. Traffic estimation based receiver initiated mac for underwater acoustic networks. In *Proceedings of ACM WUWNet*, pages 1–7. ACM, 2013.

[89] National Oceanic and Atmospheric Administration. National data buoy center. ndbc.noaa.gov. [Oneline]. Available: `http://www.ndbc.noaa.gov/station_history.php?station=44066`. [Accessed: April, 2014].

[90] R. Jain, A. Durresi, and G. Babic. Throughput fairness index: An explanation. Technical report, Tech. rep., Department of CIS, The Ohio State University, 1999.

[91] T. Ho, M. Médard, R. Koetter, D.R. Karger, M. Effros, J. Shi, and B. Leong. A random linear network coding approach to multicast. *IEEE Transactions on Information Theory*, 52(10), 2006.

[92] Hai-Heng Ng, Wee-Seng Soh, and Mehul Motani. ROPA: A MAC protocol for underwater acoustic networks with reverse opportunistic packet appending. In *Proceedings of Wireless Communications and Networking Conference (WCNC)*, pages 1–6. IEEE, 2010.

[93] Yu Luo, Lina Pu, Zheng Peng, Yibo Zhu, and Jun-Hong Cui. RISM: An efficient spectrum management system for underwater cognitive acoustic networks. In *Proceedings of International Conference on Sensing, Communication, and Networking (SECON)*, pages 414–422. IEEE, 2014.

[94] Nitthita Chirdchoo, Wee-Seng Soh, and Kee Chaing Chua. RIPT: a receiver-initiated reservation-based protocol for underwater acoustic networks. *IEEE Journal on Selected Areas in Communications*, 26:1744–1753, 2008.

[95] J.J. Garcia-Luna-Aceves and Asimakis Tzamaloukas. Reversing the collision-avoidance handshake in wireless networks. In *Proceedings of International Conference on Mobile Computing and Networking (MobiCom)*. ACM, 1999.

[96] Yibo Zhu, Son Le, Lina Pu, Xiaoyan Lu, Zheng Peng, Jun-Hong Cui, and Michael Zuba. Aqua-Net Mate: a real-time virtual channel/modem simulator for Aqua-Net. In *Proceedings of IEEE OCEANS*, Bergen, Norway, June 2013.

[97] Chee-Yee Chong and S. P. Kumar. Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003.

[98] Haining Mo, Zhong Zhou, Michael Zuba, Zheng Peng, Jun-Hong Cui, and Yantai Shu. Practical coding-based multi-hop reliable data transfer for underwater acoustic networks. In *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, December 2012.

[99] Christian Ritter and Martin A. Tanner. Facilitating the gibbs sampler: The gibbs stopper and the griddy-gibbs sampler. *Journal of the American Statistical Association*, 87(419):pp. 861–868, 1992.

[100] Feng Lu, Diba Mirza, and Curt Schurgers. D-Sync: doppler-based time synchronization for mobile underwater sensor networks. In *Proceedings of ACM WUWNet*, 2010.

[101] Jun Liu, Zhaohui Wang, Zheng Peng, Michael Zuba, Jun-Hong Cui, and Shengli Zhou. TSMU: A time synchronization scheme for mobile underwater sensor networks. In *Proceedings of Global Telecommunications Conference (GLOBECOM)*,

pages 1–6. IEEE, 2011.

[102] Zhengbao Li, Zhongwen Guo, Feng Hong, and Lu Hong. E^2DTS: an energy efficiency distributed time synchronization algorithm for underwater acoustic mobile sensor networks. *Ad Hoc Networks*, 11(4):1372–1380, 2013.

[103] Martin Horauer, Klaus Schossmaier, Ulrich Schmid, Roland Höller, and Nikolaus Kerö. PSynUTC – evaluation of a high-precision time synchronization prototype system for ethernet LANs. In *Proceedings of the 34th Annual Precise Time and Time Interval Meeting*, 2002.

[104] Jason L Hill and David E Culler. Mica: a wireless platform for deeply embedded networks. *IEEE Micro archive*, 22(6):12–24, 2002.

[105] Michael A Tope and John C McEachen. Unconditionally secure communications over fading channels. In *Proceedings of IEEE Military Communications Conference*, volume 1, pages 54–58, 2001.

[106] Babak Azimi-Sadjadi, Aggelos Kiayias, Alejandra Mercado, and Bulent Yener. Robust key generation from signal envelopes in wireless networks. In *Proceedings of ACM conference on Computer and Communications Security (CCS)*, pages 401–410, 2007.

[107] ST-B Hamida, J-B Pierrot, and Claude Castelluccia. An adaptive quantization algorithm for secret key generation using radio channel measurements. In *Proceedings of International Conference on New Technologies, Mobility and Security (NTMS)*, pages 1–5. IEEE, 2009.

[108] Milica Stojanovic and James Preisig. Underwater acoustic communication channels: Propagation models and statistical characterization. *IEEE Communications Magazine*, 47(1):84–89, 2009.

[109] Parastoo Qarabaqi and Milica Stojanovic. Statistical modeling of a shallow water acoustic communication channel. In *Proceedings of Underwater Acoustic Measurements Conference*, 2009.

[110] Sriram Nandha Premnath, Suman Jana, Jessica Croft, Prarthana Lakshmane Gowda, Mike Clark, Sneha Kumar Kasera, Neal Patwari, and Srikanth V Krishnamurthy. Secret key extraction from wireless signal strength in real environments. *IEEE Transactions on Mobile Computing*, 12(5):917–930, 2013.

[111] Gilles Brassard and Louis Salvail. Secret-key reconciliation by public discussion. In *Proceedings of the Eurocrypt on Advances in Cryptology*, pages 410–423. Springer, 1994.

[112] Charles H Bennett, Gilles Brassard, Claude Crépeau, and Ueli M Maurer. Generalized privacy amplification. *IEEE Transactions on Information Theory*, 41(6):1915–1923, 1995.

[113] Xavier Lurton. *An introduction to underwater acoustics: principles and applications*. Springer, 2002.

[114] Cecil W Farrow. A continuously variable digital delay element. In *Proceedings of IEEE International Symposium on Circuits and Systems*, pages 2641–2645. IEEE, 1988.

[115] Kari Karhunen. *Über lineare Methoden in der Wahrscheinlichkeitsrechnung*, volume 37. Universitat Helsinki, 1947.

[116] Michel Loeve. *Probability theory*, volume 2. Springer, 4 edition, 1978.

[117] Qiang Huang, Johnas Cukier, Hisashi Kobayashi, Bede Liu, and Jinyun Zhang. Fast authenticated key establishment protocols for self-organizing sensor networks. In *Proceedings of Wireless Sensor Networks and Applications*, pages 141–150. ACM, 2003.

[118] Duncan S Wong and Agnes H Chan. Mutual authentication and key exchange for low power wireless communications. In *Proceedings of MILCOM*, volume 1, pages 39–43. IEEE, 2001.

[119] Li Wei, Zheng Peng, Hao Zhou, Jun-Hong Cui, Shengli Zhou, Zhijie Shi, and James O'Donnell. Long island sound testbed and experiments. In *Proceedings of IEEE OCEANS*, pages 1–6, San Diego, California, September 2013.

[120] University of Connecticut. Monitoring your sound. mysound.uconn. edu. [Oneline]. Available: `http://mysound.uconn.edu/apmet_hx.html`. [Accessed: November, 2013 - April, 2014].

[121] Andrei N Kolmogorov. Three approaches to the quantitative definition of information. *Problems of information transmission*, 1(1):1–7, 1965.

[122] Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, and Elaine Barker. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical report, DTIC Document, 2001.