

**TRANSMISSION SCHEDULING STRATEGIES  
FOR UNDERWATER ACOUSTIC NETWORKS  
WITH LARGE PROPAGATION DELAYS**

**V PRASAD ANJANGI**

*(B.E.), Andhra University*

*(M.Tech.), Indian Institute of Technology, Bombay*

**A THESIS SUBMITTED**

**FOR THE DEGREE OF DOCTOR OF PHILOSOPHY  
DEPARTMENT OF ELECTRICAL & COMPUTER  
ENGINEERING  
NATIONAL UNIVERSITY OF SINGAPORE**

**2016**



## DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

---

V Prasad Anjani

22<sup>nd</sup> December 2016



## ACKNOWLEDGEMENTS

First and foremost I would like to thank my supervisor, Prof. Mandar Chitre for providing me the opportunity to work in this field of research. Without his support and guidance, this work would not have been possible. I appreciate all his contributions from devoting his time for discussion of ideas to sharing his expertise in performing quality research. Working with him has made my research experience productive and stimulating. The enthusiasm he has for research has instilled in me energy and joy as I leave motivated to carry out impactful research after the culmination of this PhD thesis. I would also like to thank the examiners and the anonymous reviewers of the journal and conference papers for providing inputs which have significantly improved the quality of the work presented in this thesis.

The members of Acoustic Research Laboratory (ARL) have contributed immensely to my professional and personal time spent during the period of this research. I would like to thank Ms. Lee Lin for helping me out with proofreading the research papers. I also thank Mr. Too Yuen Min for healthy discussions which have helped me immensely in this research. Special thanks goes to Mr. Kee, Mr. Willie, Mr. Gabriel and Mr. Anshu for the help provided during the experiments carried out at sea. I would also like to thank Dr. Venu for providing the opportunity to carry out the experiments at sea which resulted in part of the experimental contribution presented in this thesis. My colleagues Dr. Mansoor, Dr. Ahmed, Dr. Hari and Mr. Rajat have been excellent companions over the discussions both formal and informal and I would like to express my gratitude

for giving useful inputs at appropriate times during the course of this research.

Special recognition goes to my loving parents for their support and unconditional love. To my brother and sister-in-law and my wonderful little niece Mokshita for adding joy and happiness in my life. Without them none of this work would have been possible. Lastly, special thanks goes to my friend Sudha who has always been there through the ups and downs and has played an immense role in making through this far.

# Contents

---

<b>Abstract</b>	<b>iii</b>
<b>List of Acronyms</b>	<b>xii</b>
<b>List of Notation</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Thesis Goals . . . . .	4
1.3 Thesis Contributions . . . . .	5
1.4 Thesis Outline . . . . .	7
<b>2 Literature Review</b>	<b>9</b>
2.1 MAC Protocols Mitigating Large Propagation Delays . . . . .	9
2.2 MAC Protocols Exploiting Large Propagation Delays . . . . .	11
2.2.1 State-of-the-art slotted TDMA based strategies . . . . .	14
2.2.2 State-of-the-art unslotted TDMA based strategies . . . . .	15
2.2.3 Open problems . . . . .	16
2.3 Preliminary Concepts and Definitions . . . . .	17
<b>3 Slotted Schedules with Transmission Power Control</b>	<b>22</b>
3.1 System Model and Assumptions . . . . .	23
3.2 Scheduling Problem . . . . .	25
3.2.1 Choosing time slot length . . . . .	25
3.2.2 Link scheduling optimization problem . . . . .	26
3.3 Performance Evaluation . . . . .	35
3.3.1 Geometries found with throughput gain . . . . .	37
3.3.2 Simulation results . . . . .	42
3.4 Summary . . . . .	45
<b>4 Unslotted Schedules with Variable Packet Lengths</b>	<b>47</b>
4.1 System Model and Assumptions . . . . .	48
4.2 Problem Formulation . . . . .	50
4.2.1 Propagation delay constraints . . . . .	53
4.2.2 Duration between two consecutive transmissions . . . . .	54
4.2.3 Allowing transmissions and receptions across frame boundary . . . . .	55
4.2.4 Throughput . . . . .	55
4.2.5 Inclusion of packet headers . . . . .	55
4.2.6 Objective function . . . . .	56
4.3 MILFP Solution . . . . .	58
4.4 Results . . . . .	59
4.4.1 Throughput gain - Sea-trial network geometry . . . . .	61

4.4.2	Throughput gain - Randomly deployed network geometries	72
4.5	Network Geometries with Arbitrary Traffic Demands . . . . .	73
4.5.1	Illustrative network geometries . . . . .	74
4.6	Scalability & Complexity Analysis . . . . .	81
4.7	Region of Operation - Exploiting Large Propagation Delays . . .	85
4.7.1	Simulation results . . . . .	87
4.8	Summary . . . . .	89
<b>5</b>	<b>Robust &amp; Unslotted Schedules for Practical Multihop Grid Networks</b>	<b>91</b>
5.1	System Model and Assumptions . . . . .	94
5.2	Time-Slotted $\rho$ -Schedules for Multihop Networks . . . . .	97
5.3	Problem Formulation . . . . .	100
5.3.1	Propagation delay constraints . . . . .	102
5.3.2	Allowing transmissions and receptions across frame boundary . . . . .	104
5.3.3	Throughput . . . . .	104
5.3.4	Objective function . . . . .	104
5.3.5	Robust MILFP formulation . . . . .	109
5.4	Results . . . . .	111
5.4.1	Unslotted schedules . . . . .	112
5.4.2	Robust unslotted schedules . . . . .	120
5.5	Summary . . . . .	124
<b>6</b>	<b>Modem Constraints and Experimental Demonstration</b>	<b>126</b>
6.1	Transition Times in Modem . . . . .	127
6.2	Optimization Problem . . . . .	129
6.2.1	Without modem constraints . . . . .	129
6.2.2	With modem constraints . . . . .	130
6.3	Case Study . . . . .	131
6.3.1	Results . . . . .	132
6.4	Implementation & Experimental Demonstration . . . . .	137
6.4.1	Practical Super-TDMA schedule . . . . .	138
6.4.2	Overview of the implementation . . . . .	141
6.4.3	Experimental results . . . . .	144
6.5	Summary . . . . .	145
<b>7</b>	<b>Discussion, Conclusions &amp; Future Research</b>	<b>146</b>
7.1	Discussion & Conclusions . . . . .	146
7.2	Future Research . . . . .	151
	<b>Bibliography</b>	<b>154</b>
	<b>List of Publications</b>	<b>162</b>

## Abstract

---

Propagation delays in underwater acoustic (UWA) networks are large when compared to those in radio-frequency based terrestrial wireless networks due to the slower speed of sound. Several Medium Access Control (MAC) protocol designs for UWA networks have attempted to mitigate its ill-effects. However, studies on the fundamental understanding of large propagation delays have highlighted the advantages of its exploitation rather than mitigation in the design of throughput-maximizing MAC strategies. A review of the state-of-the-art in the development of MAC protocols utilizing the propagation delay information reveals open problems such as: (a) how to compute throughput-maximizing transmission schedules for arbitrarily deployed practical UWA networks with packet traffic demands; (b) how to design transmission strategies that better exploit large propagation delays than the current state-of-the-art techniques; (c) how to extend such techniques to much larger multi-hop networks; (d) how to compute transmission schedules robust to the uncertainties in the propagation delay information; and (e) whether such techniques are implementable in practice on underwater acoustic modems, and what are the practical challenges in the implementation. This thesis contributes toward answering these problems and facilitates the advancement of such novel techniques one step closer to reality through experimental demonstration.

This research presents strategies to improve network throughput in arbitrary

networks by better utilization of propagation delay information leading to a network throughput much closer to the upper bound. The first strategy to improve throughput in arbitrary network geometries adopts time-slotted schedules. Given a suitable network geometry, it has been shown that the network throughput increases linearly with the number of nodes, provided optimal transmission schedules utilizing propagation delays are adopted. What, though, about arbitrary network geometries? The throughput of an arbitrary network geometry is often more constrained, although it can be increased by controlling the transmission power which assists in reducing energy consumption and limits interference. This research investigates achievable throughput in randomly deployed underwater acoustic networks by controlling the transmission power. It develops a joint scheduling and power control algorithm for arbitrary network geometries. Performance improvement is demonstrated for a large number of random network geometries. Some network geometries are also presented for which optimal throughput is achieved by using transmission schedules with power control. Although power control limits the interference and thereby provides more transmission opportunities, the solution is still time-slotted with fixed packet lengths.

A packet can be divided into several smaller packets of different sizes. The packet duration provides a degree of freedom that, if utilized, results in strategies that can be adopted to achieve throughput closer to the established upper bound. Therefore as a second strategy, this research considers the unslotted transmission of packets with arbitrary size. Given the propagation delay between nodes in the network and packet traffic demands, an optimization problem is formulated

for minimizing the fractional idle time in a frame (or period) of the schedule as a Mixed-Integer Linear Fractional Problem (MILFP). The results are compared to those from recent works that also exploit large propagation delays using either time-slotted or unslotted schedules with fixed packet duration. Also presented are schedules which have been computed for various network geometries with arbitrary packet traffic demands.

The research is also extended to much larger multi-hop grid networks. A recent work investigated a regular grid-based network topology with multi-hop relaying. A transmission strategy which maximizes the throughput while exploiting the large propagation delay was presented, and the upper bound on throughput established. Two practical problems are identified by generalizing the assumptions. Firstly, when communication nodes are deployed in the ocean, there will inevitably be slight positional deviations in the actual locations of the nodes from the perfectly aligned regular grid networks assumed. Secondly, there exists uncertainty in propagation delays. These uncertainties arise due to varying ocean currents as well as changes in the ocean's physical parameters. Both the irregularity in the grid network due to deployment errors, and the uncertainties in propagation delays among the nodes, significantly degrade the network throughput. This research formulates a robust scheduling problem as an MILFP and computes throughput-maximizing robust schedules which are per link fair. It also demonstrates the throughput gain compared to the state-of-the-art and verifies the robustness of the solution to uncertainties in propagation delays.

Theoretical research has made advances in the study of the impact of

large propagation delays in UWA networks and many complex techniques have been proposed for interference management in order to maximize the network throughput. However, a gap exists between the theoretically established gains and the achievable gains in practice. This research studies the practical modem constraints and implements such schedules on the underwater acoustic modem. The inclusion of modem constraints results in the design which can be realized in practice. The optimal packet and time slot lengths are computed, resulting in the maximum utilization of time slots while minimizing the guard times. Through experiments in Singapore waters, the research demonstrates interference alignment, the crossing of simultaneously transmitted packets in water, and the time synchronization among the deployed nodes in the network.

## List of Tables

---

2.1	Schedule for an equilateral triangle network . . . . .	20
4.1	Time-Slotted fixed packet duration solution . . . . .	63
4.2	Unslotted fixed packet duration solution . . . . .	65
4.3	Variable packet duration solution . . . . .	69
4.4	Variable packet duration solution with packet headers . . . . .	69
4.5	Arbitrary packet traffic demands for illustrative network geometries considered . . . . .	75
4.6	Optimal schedules for equilateral triangle network geometry . . . . .	75
4.7	Optimal schedules for isosceles triangle network geometry . . . . .	76
4.8	Optimal schedules for linear network geometry . . . . .	76
4.9	Link propagation delays . . . . .	88
5.1	$\rho$ -Schedule . . . . .	117
5.2	Unslotted schedule . . . . .	118
5.3	Robust unslotted schedules . . . . .	123
6.1	Solution without modem constraints . . . . .	132
6.2	Feasible solution with modem constraints . . . . .	134
6.3	Optimal solution with modem constraints . . . . .	135
6.4	Transmission schedule for 3 node linear network . . . . .	138
6.5	Distance matrix of the sea-trial 3 node network . . . . .	138

## List of Figures

---

3.1	5 node trivial random network geometry with throughput $S = 2.5$ and $\rho$ -throughput $S_\rho = 1.99$ . . . . .	37
3.2	4 node non-trivial network geometry with throughput $S = 2$ and $\rho$ -throughput $S_\rho = 2$ . . . . .	39
3.3	5 node non-trivial random network geometry with throughput $S = 2.5$ and $\rho$ -throughput $S_\rho = 1.45$ . . . . .	39
3.4	Throughput values showing 95% confidence interval is plotted as a function of the number of nodes in the network. . . . .	43
3.5	Percentage increase in average throughput due to power control. . . . .	44
3.6	Average & Maximum throughput over 200 random network geometries. . . . .	44
3.7	Histogram of throughput with power control for 200 randomly generated 5 node networks. . . . .	45
4.1	Illustration of the effect of propagation delay on transmission times of packets to be transmitted on links $(j, k)$ and $(l, i)$ . . . . .	50
4.2	UNET network node locations during the MISSION 2013 experiment (deployment #1). White markers are network nodes. The geometry considered for the case study is marked with the distances between the links. . . . .	60
4.3	Throughput sensitivity to time slot length while using the algorithm in [1] and to packet duration while using the algorithm in [2]. . . . .	65
4.4	Throughput comparison when using proposed MILFP technique resulting in unslotted variable packet duration schedules for the considered sea-trial network geometry. . . . .	66
4.5	Schedule visualization with fixed time slot length and packet duration. The guard times set in the time slot take care of the early receptions and delayed receptions problem due to the approximations made in the delay matrix. Throughput achieved is 1.35. . . . .	67

4.6	Schedule visualization for unslotted fixed packet duration schedule. Note that all the packet transmissions are of equal size - 539 ms. Throughput computed is 1.32. . . . .	67
4.7	Schedule visualization with variable packet duration. Note that there are no time slots and the transmission times and packet lengths are such that the total idle time is minimized. Throughput achieved is 1.484, which is significantly closer to the upper bound on the throughput. . . . .	68
4.8	Average throughput is computed over 100 network geometries with 3 nodes and 6 links to compare against the known upper bound on the throughput. . . . .	71
4.9	Regular N-node multi-line grid network with uniformly spaced nodes. . . . .	82
4.10	Scalability with number of nodes and links in a multi-hop multi-line grid network using MILFP. . . . .	83
4.11	Computation time with number of nodes and links in a multi-hop multi-line grid network using MILFP. . . . .	83
4.12	Sea-trial network geometry. . . . .	86
4.13	Isosceles triangle network geometry. . . . .	87
5.1	Practical N-node multi-line grid network with unevenly spaced nodes. . . . .	94
5.2	The figure shows the nodes lying in the collision domain associated with node $j$ 's transmission to node $k$ . The list of these nodes are enumerated in the set $\mathcal{I}_{jk}$ . Note that due to the slight irregularity in the grid network, the number of nodes lying in the interference range associated with each link $(j, k)$ can be different. This is not the case in regular grid networks considered in [3]. . . . .	98
5.3	Propagation delay uncertainty set considering the worst case scenarios. . . . .	108
5.4	Average throughput is computed over 100 random instances of grid networks for different number of nodes. . . . .	112
5.5	Throughput is plotted as the interference range increases causing more nodes in the adjacent lines to interfere with the transmissions. . . . .	114

5.6	Average throughput is computed over 200 random deployments of 12-node 3-line multi-hop grid network. The irregularity in the network is introduced by varying the radius of the Euclidean ball in which the nodes are randomly deployed. . . . .	115
5.7	Throughput computed for each of the 200 random deployments at each of the selected radius $r$ is shown as a notched box plot to demonstrate that the true medians do differ from each other with 95 % confidence. . . . .	116
5.8	Verification of unslotted and time-slotted schedules on UNET simulator for 12-node 3-line multi-hop grid network. . . . .	117
5.9	Two different grid network deployment scenarios are considered. For each deployment, the uncertainties in the propagation delay are induced. 100 random instances with $r_{\max} = 0.2$ and $0.25$ are considered. The average throughput is computed and plotted against the randomness factor $\gamma$ . . . . .	121
6.1	Schedule visualization without modem constraints (the guard times set in the time slot take care of only the early receptions and delayed receptions problem due to the approximations made in the delay matrix because of the network geometry). . . . .	133
6.2	Schedule visualization with modem constraints (the guard times set in the time slot not only take care of the constraints due to geometry but also consider TX-RX and RX-TX transition times in modem). . . . .	134
6.3	Throughput comparison of Traditional-TDMA and Super-TDMA protocols with and without modem constraints for the considered network geometry. . . . .	136
6.4	UNET network node locations during the experiment in Singapore waters. Yellow markers are network nodes. The links considered for demonstrating Super-TDMA are marked with distances between them. . . . .	139
6.5	Overview of the implementation showing the modules: synchronizer and scheduler. Both the modules are implemented on all 3 nodes with the objective to first achieve the time synchronization among the nodes and second to prepare the modems for accurately transmit and receive at scheduled times. .	139

6.6	Visualization of the expected scheduled events in time is shown. Note the crossing of the transmitted packets in the medium, interference alignment in the transmitting slots, zero interference during receiving slots and the scheduler job sequence to configure the modems to switch between transmission and reception modes at appropriate times. . . . .	140
6.7	Visualization of the actual events as happened at sea in time is shown. Note the crossing of the transmitted packets in the medium. The intended overlap of the interfering packets happened as expected in the corresponding time slots. The time synchronization is achieved perfectly as can be observed by looking at the first packets that are transmitted on all three nodes.	142
7.1	Effect of spreading delay on Super-TDMA schedules. . . . .	149

## List of Acronyms

---

UWA	Underwater Acoustic
MAC	Medium Access Control
TDMA	Time Division Multiple Access
FDMA	Frequency Division Multiple Access
CDMA	Code Division Multiple Access
MACA	Multiple Access with Collision Avoidance
ILP	Integer Linear Problem
MILP	Mixed-Integer Linear Problem
MILFP	Mixed-Integer Linear Fractional Problem
STUMP	Staggered TDMA Underwater MAC Protocol
APCAP	Adaptive Propagation-Delay Tolerant Collision Avoidance Protocol
R-MAC	Reservation based MAC
MACA-U	MACA for Underwater
RIPT	Receiver Initiated Reservation Protocol
RTS	Request to Send
CTS	Clear to Send
ARL	Acoustic Research Laboratory
UNET-II	ARL Underwater Network Modem
OCCO	Oven Controlled Crystal Oscillator

## List of Notation

---

$(j, k)$	A directed link with node $j$ as transmitter and node $k$ as receiver
$(l, i)$	A directed link with node $l$ as transmitter and node $i$ as receiver
$\mathcal{L}$	Set of all possible links in the network
$\mathcal{L}_s$	Set of all links to be scheduled in the network
$D_{jk}$	Propagation delay corresponding to link $(j, k)$
$\hat{D}_{jk}$	Uncertain propagation delay corresponding to link $(j, k)$
$\mathcal{D}_{jk}$	The set containing all possible propagation delay values corresponding to link $(j, k)$
$\Lambda_{jk}$	The number of packets to be transmitted on link $(j, k)$ in a single frame
$t_{jk}^x$	Transmission time of the $x^{\text{th}}$ packet scheduled on link $(j, k)$
$t_{li}^y$	Transmission time of the $y^{\text{th}}$ packet scheduled on link $(l, i)$
$\tau_{jk}^x$	Transmission/Packet duration of the $x^{\text{th}}$ packet scheduled on link $(j, k)$
$\tau_{li}^y$	Transmission/Packet duration of the $y^{\text{th}}$ packet scheduled on link $(l, i)$
$t_p$	Packet header duration
$\tau_{lb}$	Lower bound on the packet duration on all links
$\alpha$	Ratio of interference range to communication range
$p_{jk,li}^{xy}, q_{jk,li}^{xy}, r_{jk,li}^{xy}$	Binary variables associated with each pair of variables $t_{jk}^x$ and $t_{li}^y$ for the Big-M transformation
$T$	Frame duration

$S$	Normalized network throughput
$S_\rho$	Normalized network throughput computed using $\rho$ -Schedule
$\beta$	Number of adjacent frames in the past and future considered for collisions due to propagation delay
$G$	Propagation delay corresponding to the maximum interference range in the network
$\mathbf{W}^{(T)}$	Schedule matrix with period $T$
$\tau$	Time slot length
$t_s$	Guard time at the start of time slot
$t_e$	Guard time at the end of time slot
$t_{\text{RX-TX}}$	Delay to switch from RX mode to TX mode
$t_{\text{TX-RX}}$	Delay to switch from TX mode to RX mode
$t_{\text{TX-TX}}$	Delay to transmit a packet if already in TX mode
$t_{\text{RX-RX}}$	Delay to receive a packet if already in RX mode
$\Delta x$	Smallest incremental packet duration in modem
$\eta$	Slot utilization efficiency
$\tau_{\min}$	Minimum time slot length
$\tau_{\max}$	Maximum time slot length
$t_{\text{payload}}$	Payload duration
$t_{\text{pd}}$	Packet duration including packet header and payload
$t_{\text{maxpd}}$	Maximum propagation delay among the links links considered in the network
$\gamma$	Randomness factor
$B(\mathbf{x}, r)$	Radius of Euclidean ball centered at location $\mathbf{x}$ with a radius $r$

# Chapter 1

## Introduction

---

### 1.1 Motivation

The oceans cover 70% of the Earth's surface and is not easily accessible. There is a growing need for developing applications that can support oceanic research and exploration as well as those which can provide the commercial advantage in submarine operations. Underwater communications and networking field is growing rapidly due to the increasing impact it has in such applications. Environmental monitoring, disaster prevention, offshore exploration, tactical surveillance, etc. are among some of the applications where underwater networks can play a major role [4]–[43]. Military applications such as autonomous sensor networks for anti-submarine warfare are also emerging [39], [44], [45]. The communication among underwater assets is an essential component of these applications. Radio-frequency waves do not propagate well in water and suffer from high attenuation [46]. Optical waves do not suffer from such high attenuation but are affected by scattering. The communication ranges that can be achieved through underwater optical communication are in the range of few tens of meters [47], [48]. Acoustic communication is the choice of transmission technology for most underwater networked systems and provides reasonable bandwidth along with few kilometers of communication range [49].

The Underwater Acoustic (UWA) channel has unique characteristics like severe transmission loss, time-varying multipath propagation, doppler-spread, limited bandwidth and slow propagation speed of sound resulting in high and variable propagation delay [50], [51]. In most terrestrial wireless systems using radio-frequency waves, the propagation delay is small compared to the packet size and allows to effectively deal with the ill-effects of propagation delays. The propagation speed of acoustic waves underwater is about 1500 m/s, which is several orders of magnitude slower than the speed of radio-frequency waves, i.e.,  $3 \times 10^8$  m/s. Therefore, the propagation delays are non-negligible in UWA networks, unlike terrestrial wireless networks. Due to the presence of non-negligible propagation delays, Medium Access Control (MAC) protocols designed for terrestrial wireless networks cannot be directly adopted for UWA networks [32], [52]. Efficient protocol design for UWA networks in the presence of such unique challenges is required across different layers of networking protocol stack.

At the MAC layer, efficient scheduling and resource allocation are essential components for enabling throughput-maximization in wireless networks adopting contention-free protocols. The class of contention-free protocols includes Code Division Multiple Access (CDMA), Frequency Division Multiple Access (FDMA) and Time Division Multiple Access (TDMA). These protocols divide the available channel capacity in the acoustic channel into a set of logical channels for the purpose of multiple-access. Among these protocols, FDMA is considered inefficient for underwater applications [53]. Most of the work has focused on CDMA and TDMA based protocols. CDMA based MAC protocols with

power control have been proposed for underwater networks [54] and have the advantages of being robust to multi-path fading and for no requirement of time synchronization. However, in the case of CDMA, the available physical layer resources, such as bandwidth and power must be divided among the users. An advantage of TDMA based protocols is that they provide flexibility in terms of implementation over any physical layer technology.

In traditional TDMA, collision-free reception is guaranteed by allocating a particular time slot to each transmitting node. The size of the slot depends on the transmission duration plus the maximum propagation delay in the network. In UWA networks, since the propagation delays are comparable to the transmission duration, the network performance is significantly degraded using traditional TDMA-based protocols due to low channel utilization. Although huge time slots with large packet duration can be implemented in traditional TDMA, to mitigate propagation delay, it is not preferred since this will induce the large end to end delay in packet delivery. The literature on TDMA-based scheduling can be classified chronologically: earlier studies tend to mitigate the ill-effects of long propagation delays, while more recent works exploit the propagation delay in UWA networks. A fundamental understanding of the effects of large propagation delays and advantages in exploiting it to design throughput-maximizing strategies was presented in [1]. Rather surprisingly, the results in [1] show that by exploiting large propagation delays, optimal schedules can be computed for networks which perform better than their counterparts in terrestrial wireless networks. Motivated by the findings in [1], we focus on studying the strategies that can exploit long propagation delays in arbitrary

UWA networks. We study the strategies which can perform better than the current state-of-the-art techniques and also present the practical challenges in implementing such solutions on underwater acoustic modems.

This thesis presents a study on the design of transmission scheduling strategies which can take advantage of large propagation delays in UWA networks. The objective of rest of the chapter is to provide an overview of this thesis and state the specific objectives and contributions.

## 1.2 Thesis Goals

The aims of this thesis can be succinctly summarized as follows:

1. To propose a centralized algorithm which can better utilize the propagation delay information in arbitrary UWA networks when compared to the current state-of-the-art centralized algorithms.
2. To propose a centralized algorithm which can better utilize the propagation delay information while maintaining fairness among the links as required in multi-hop multi-line grid networks.
3. To present the scalability of the algorithm exploiting large propagation delays to much larger practical multi-hop multi-line grid networks.
4. To design an algorithm that can compute schedules which are robust to the uncertainties in the propagation delay information.
5. To model the practical constraints in underwater acoustic modems, find transmission schedules which are implementable and present experimental demonstration of the technique at sea.

### 1.3 Thesis Contributions

Our work primarily focuses on the design and implementation of a MAC layer technique responsible for assigning the transmission start times and transmission duration for all the links considered for scheduling. The specific contributions of this thesis are summarized as follows:

1. We investigate achievable throughput in randomly deployed UWA networks by controlling transmission power in case of time-slotted schedules. We develop joint scheduling and power control algorithm for arbitrary networks and demonstrate performance improvement in a large number of random network geometries.
2. We formulate a TDMA based scheduling problem as a Mixed-Integer Linear Fractional Problem (MILFP) which better exploits long propagation delays in static UWA networks. We consider minimizing the total fractional idle time in a frame which allows for multiple concurrent transmissions in the medium.
3. We demonstrate that MILFP results in unslotted and variable packet length schedules and performs better than other state-of-the-art centralized algorithms available for computing optimal schedules. We compare the proposed strategy to the following techniques:
  - (a) The time-slotted fixed packet duration schedules that are computed using centralized algorithm in [1].
  - (b) The unslotted fixed packet duration schedules that are computed

using centralized algorithm in [2].

4. We consider some network geometries for which the optimal schedules are known from [1] and present the schedules when arbitrary packet traffic demands are to be satisfied on each link. This demonstrates the capability of the proposed algorithm to compute transmission schedules with different packet traffic demands.
5. The large propagation delay is relative to the transmitted packet duration and hence we quantify the values and show the regions in which the networks must be operated in order to achieve significant throughput gains when compared to radio-frequency based terrestrial wireless networks with negligible propagation delays.
6. We demonstrate the throughput gains that can be achieved in practical multi-hop multi-line grid networks using unslotted schedules and compare this to the state-of-the-art time-slotted schedules presented in [3]. We derive the necessary condition for the proposed algorithm to always perform better than the time-slotted solution.
7. To deal with the uncertainties in the propagation delay values that can arise because of the slight drift of the nodes due to varying ocean currents as well as changes in the oceans physical parameters we propose an algorithm to compute robust transmission schedules under such propagation delay uncertainties.
8. We study practical modem constraints and implement the Super-TDMA

protocol on underwater acoustic modem. The inclusion of modem constraints results in the design which can be realized in practice. The optimal packet and time slot lengths are computed which result in maximum utilization of the time slots by minimizing the guard times.

9. We deploy a 3 node network in Singapore waters and compute the schedule along with the time slot length and the guard periods and implement it on all three nodes along with time synchronization to demonstrate the key concepts at sea.

## 1.4 Thesis Outline

In Chapter 2, we present the relevant literature on the efforts in mitigating and exploiting the non-negligible propagation delays in UWA networks. We list the set of open problems which provides an understanding of the impact of our research.

In Chapter 3, we consider the time-slotted schedules that can be computed using a state-of-the-art algorithm presented in [1] and study the effect of limiting the interference range through transmission power control on network throughput for random network geometries.

Chapter 4 presents an in-depth study on how to better utilize the large propagation delays in UWA networks and propose a centralized algorithm to compute unslotted and variable packet length schedules which can provide throughput-maximizing solutions for arbitrary network geometries with different packet traffic demands.

The proposed technique in Chapter 4 is extended to practical multi-hop

multi-line grid networks in Chapter 5. The uncertainties in propagation delays are formulated to compute schedules which are robust to the small variations in the propagation delays.

Having proposed strategies that in theory achieve much higher throughput than current state-of-the-art by utilizing propagation delays, we turn our attention to practical aspects of these techniques. In Chapter 6, we study the practical modem constraints and compute implementable schedules on underwater acoustic modem. We demonstrate these schedules through experiment at sea.

Chapter 7 summarizes the key findings and presents the potential future research directions.

## Chapter 2

### Literature Review

---

There have been many studies on MAC protocols for wireless sensor networks and numerous studies have pointed out that direct application of these existing MAC protocols to UWA networks leads to poor performance. The extensive reviews of such protocols are presented in [49], [55]–[57]. Our focus in this review remains on how the non-negligible propagation delays are dealt with while designing MAC protocols for UWA networks. The study reveals that the literature can be classified broadly into MAC protocols which tend to mitigate the effect of large propagation delays and MAC protocols which exploit or utilize the propagation delay information to improve channel utilization.

#### 2.1 MAC Protocols Mitigating Large Propagation Delays

As mentioned Section 1.1, traditional TDMA results in low channel utilization due to large guard times of the order of the maximum propagation delay in the network. In order to improve channel utilization, an energy efficient, distributed MAC protocol (UWAN-MAC) that works despite long, unknown propagation delays is proposed in [58]. This protocol differs from ALOHA based protocol and multiple access with collision avoidance (MACA) in that energy is the main performance metric rather than bandwidth utilization. The impact of long propagation delays on the fraction of energy wasted while using UWAN-MAC

is studied. Relative time stamps are introduced on the transmission data to allow the nodes to operate in a synchronized environment resulting in energy efficiency. The strategy reduces the ill-effects of long propagation delays and provide better efficiency through less collisions. However, in the absence of synchronization information, the network performance degrades significantly.

The need for strict time synchronization and the effects of the absence of synchronization information in UWAN-MAC from [58] is studied in [59]. The authors in [59] focus on studying the trade-off between the duty cycle of the nodes, the network throughput and the energy consumption.

In [60], the authors propose I-TDMA, a transmission scheduling strategy in which the data packets are interleaved in the empty time axis to mitigate the effect of large propagation delay in underwater channel.

A hierarchical TDMA (HTDMA) MAC protocol is proposed for clustered UWA networks in [61]. HTDMA employs two nested TDMA schedules, one is executed among the surface stations using terrestrial wireless communication and the other is executed among underwater nodes using acoustics. A strategy is proposed to enhance delay and channel utilization performance by avoiding the occurrence of idle time slots. HTDMA improves the delay performance by efficiently controlling the mobility of underwater nodes.

In [62], a distributed and robust TDMA scheduling algorithm is proposed. The information of 2-hop neighboring nodes' information is collected and is used to compute the initial time of the time slot. The set of nodes that are assigned the same initial slot form a maximal independent set. This algorithm improves the utilization of time by allowing same time slot but still large guard times are

required which are of the order of maximum propagation delay in the network.

The authors in [63] take a different approach in mitigating the effects of large propagation delays in UWA networks. A throughput-optimal, distributed random policy is studied in a single broadcast domain and a non-linear optimization problem is solved to find the optimal solution. Although the proposed technique allows for some concurrent transmissions, since a distributed algorithm with no topology information is considered, the study does not provide understanding of the full utilization of long propagation delays.

A stochastic scheduling approach similar to the one presented in [63] is presented to assign transmission probabilities that are aimed at optimizing network performance with respect to overall network latency and reliability in [64]. The authors in [64] also present heuristic objective functions for assigning transmission probabilities to improve overall network performance by mitigating the effects of long propagation delays.

## **2.2 MAC Protocols Exploiting Large Propagation Delays**

Reservation based protocols have been proposed to utilize the large propagation delays in UWA networks in several ways. The authors in [65] proposed an adaptive propagation delay-tolerant collision avoidance protocol (APCAP) that allows for concurrent multiple reservations on each node. The proposed strategy allows stations to schedule actions for multiple data frames, reducing the amount of time wasted in waiting for signals to propagate in the medium.

To improve the channel utilization by reducing the control overhead, the authors in [66] propose a reservation based MAC protocol (R-MAC). The

RTS/CTS message exchange to avoid data packet collisions are not used in R-MAC. Instead the control and data packet transmissions are scheduled at both transmitter and receiver to completely avoid the packet collisions. The authors also introduce packet bursts and delayed ACK delivery concepts to improve channel utilization.

The authors in [67] proposed MACA-U for underwater networks and replace the original MACA for benchmark comparisons with protocols proposed for underwater wireless networks. The authors modify the MACA to suit the unique characteristics of underwater channel by introducing state transition rules, packet forwarding strategy and a back-off algorithm. The packet forwarding strategy allows concurrent transmission of RTS packets on two neighboring source nodes utilizing the long propagation delays.

Receiver initiated reservation protocol (RIPT) is proposed in [68] which is a random access handshaking-based protocol that addresses the long propagation delay by utilizing receiver-initiated reservations. The receiver accept the packet transmission requests from its neighboring nodes by considering the propagation delay to its neighbors. MACA-MN is proposed by the same authors in [69] to further improve channel utilization by enabling multiple packet train to neighbors.

Since our objective is in studying the optimum utilization of propagation delays in static UWA networks, the TDMA based protocols are perfect fit for providing optimal time reuse opportunities as shown in the following studies. The authors in [70] propose a joint routing and scheduling algorithm for small underwater networks with the objective of minimizing energy consumption.

An Integer Linear Problem (ILP) is formulated and the constraints utilize the propagation delay information.

In [71], a staggered TDMA underwater MAC protocol (STUMP) is proposed. The authors show that leveraging node position diversity through scheduling yields large throughput improvements in channel utilization. The authors also explore CDMA as a physical layer access and compare it to TDMA under identical conditions concluding that CDMA although provides other benefits in underwater networks, it does not provide benefit to scheduled MAC protocols when using realistic spreading values.

STUMP-WR is proposed in [72] which extends the work in [71] by including routing in the problem formulation. A distributed routing and channel scheduling protocol is designed for heavily loaded underwater networks. STUMP-WR selects and schedules links using a distributed algorithm to overlap communications by leveraging the long propagation delays.

A time evolving conflict graph that can be utilized by centralized algorithms is proposed in [73] to schedule transmissions. The slotted spatio-temporal conflict graph considers both the packet propagation delay and link transmission delay. The authors also present the performance bounds under certain traffic loads.

In [74], an analytical model for joint MAC and routing optimization for small to medium scale networks is proposed. The work uses heuristics and considers the presence of multiple interfering nodes and the use of an underwater acoustic channel attenuation model.

Following the review of the strategies proposed over the past decade that

utilize the large propagation delays, we narrow down our focus to more recent state-of-the-art techniques as described here:

1. *Slotted TDMA based strategy*: The authors in [1] provide a fundamental understanding of the effect of large propagation delays on network throughput and propose a dynamic programming based centralized algorithm resulting in time-slotted throughput-maximizing schedules.
2. *Unslotted TDMA based strategy*: The authors in [2] propose a centralized algorithm to find unslotted schedules utilizing large propagation delays. A Mixed-Integer Linear Problem (MILP) is formulated to find the optimal schedules.

The approach followed to solve the scheduling problem in these works are very distinct. In this thesis, as we draw the insights from these works, we also present the similarities in the solution from these two distinct approaches. Next, we elucidate the work presented in [1], [2] and the closely related work before we state the open problems that exist.

### **2.2.1 State-of-the-art slotted TDMA based strategies**

The idea of allowing nodes to transmit simultaneously and letting their packets "cross in medium" has been considered before [75]–[80]. In [1], the authors present a fundamental study on what is the impact of large propagation delays in UWA networks. The authors answer the questions such as what are the achievable throughput of networks with large propagation delays and how to find optimal and near-optimal schedules for given network geometries. Specifically, the authors in [1] formulate a scheduling problem as a sequential decision problem

and solve it using dynamic programming technique. It is shown in [1] that the network throughput is upper bounded by  $\frac{N}{2}$ , where  $N$  is the number of nodes in a network. Optimal schedules achieving the  $\frac{N}{2}$  upper bound are also presented for certain network geometries. The authors also propose a method to find schedules for arbitrary network geometries but often result in throughput much lesser than upper bound  $\frac{N}{2}$ . The solution to the proposed algorithm is time-slotted and each transmission decision in the time slot is taken to maximize the future transmission opportunities. It is also observed that only the throughput-maximizing links are chosen in the schedule which maximize the transmission opportunities and thereby the network throughput. The work in [1] is extended to regular multi-hop multi-line grid networks with multiple partially overlapping collision domains in [3]. Recently, the work in [1] is also extended to adhoc linear networks with regular structure in [81].

### **2.2.2 State-of-the-art unslotted TDMA based strategies**

In [2], ST-MAC a distributed algorithm and a centralized MILP algorithm that can provide optimal schedules in presence of the large propagation delays in UWA networks is presented. The performance of the algorithm proposed in [2] is compared with S-MAC [82], T-Lohi [83] and ECDiG [84] and outperforms each one of them. A major goal of the MILP model in [2] is to account for the propagation delays in the constraints formulated. Different from the works presented in [70],[74], that focuses on minimizing energy consumption by avoiding collisions, this work aims at finding throughput-maximizing schedules by minimizing the frame length. The solution of the MILP provides the

transmission time of each link at which it has to transmit. However, note that the packet duration remains a constant in this algorithm. We term this solution to be unslotted since the transmission times do not need to necessarily lie in a particular time slot. A similar approach was also presented in [85], where a distributed algorithm for multi-hop UWA network was presented. However the model formulating the propagation delay constraints presented in this work do not consider the interframe and intraframe constraints as considered in [2].

### 2.2.3 Open problems

We state the following open problems arising from the literature review:

1. For time-slotted schedules using algorithm proposed in [1], what is the effect of limiting the interference range through transmission power control ? Does it provide schedules with larger throughput ?
2. Guard times are needed in the time-slotted schedules computed for arbitrary network geometries to avoid packet collisions [1]. This degrades the network throughput significantly. How can we reduce the idle time due to the inclusion of such guard times ?
3. The solution in [1] results in scheduling only those links which maximize the throughput. How to find transmission schedule when a set of particular links needs to be scheduled ? Moreover, how to deal with the packet traffic demands on each of those links ?
4. The multi-hop multi-line grid networks considered in [3] have a regular structure which is often not the case when deployed in reality. How does

it effect the throughput ?

5. The authors in [1] and [2] formulate the scheduling problem using different methodologies. What are the similarities and differences in the solutions computed using these methods ?
6. What are the practical challenges in implementing such schedules exploiting large propagation delays on underwater acoustic modems ?

### 2.3 Preliminary Concepts and Definitions

In this thesis, we utilize the results presented in [1] and compare them against the proposed strategies. We also extend the algorithm presented in [1] by limiting the interference range to create more transmission opportunities. We present some preliminary concepts and definitions from [1] that are useful in understanding the technique we propose in Chapter 3 and also the comparisons that we draw in rest of the chapters.

#### **Integer & non-integer delay matrices**

Consider an  $N$  node network with the location of node  $i$ ,  $i \in \{1, \dots, N\}$  defined by the coordinates represented in a vector  $\mathbf{x}_i$ . The network geometry formed by the  $N$  nodes can be represented in the form of a delay matrix as shown in [1], where each element of the delay matrix contains the propagation delay between the corresponding pair. We denote the delay matrix by  $\mathbf{D}$  and the elements of  $\mathbf{D}$  are written as:

$$D_{ij} = \frac{|\mathbf{x}_i - \mathbf{x}_j|}{c\tau}, \quad i, j \in \{1, 2, \dots, N\} \quad (2.1)$$

where  $c$  is the speed of sound underwater and  $\tau$  is the time slot length. It is important to note that the elements of the delay matrix are propagation delays between nodes in units of time slot length  $\tau$  and can be rational numbers, i.e.,  $\mathbf{D}$  can be a non-integer delay matrix. But with appropriate choice of time slot length  $\tau$ , the given non-integer delay matrix can be approximated by an integer delay matrix  $\mathbf{D}'$  [1]:

$$D'_{ij} = \left\lceil \frac{|\mathbf{x}_i - \mathbf{x}_j|}{c\tau} \right\rceil, \quad i, j \in \{1, 2, \dots, N\} \quad (2.2)$$

where by  $\lceil a \rceil$  we denote the closest integer to the real value  $a$ .

The difference between the corresponding elements of the integer and non-integer delay matrix provides us information that can be used to select the time slot length that enables efficient utilization of slots. The maximum packet length that can be transmitted in a time slot with length  $\tau$ , given  $\rho^+$  and  $\rho^-$  is  $\tau(1 - \rho^- - \rho^+)$  [1], where  $\rho^+$  and  $\rho^-$  are given by:

$$\rho^+ = \max_{ij} (D_{ij} - D'_{ij}) \quad (2.3)$$

$$\rho^- = -\min_{ij} (D_{ij} - D'_{ij}) \quad (2.4)$$

where  $i, j \in \{1, 2, \dots, N\}$  for a fully-connected network.

### Schedule matrix

A schedule is denoted by matrix  $\mathbf{W}$  which determines the time slots in which each node in the network transmits and receives messages.  $W_{j,t}$  represents the element at the  $j^{\text{th}}$  row and  $t^{\text{th}}$  column of the matrix  $\mathbf{W}$ . It can be elucidated as

follows:

1. If  $W_{j,t} = i > 0$ , then node  $j$  transmits a message to node  $i$  in time slot  $t$ .
2. If  $W_{j,t} = -i < 0$ , then node  $j$  receives a message from node  $i$  in time slot  $t$ .
3. If  $W_{j,t} = 0$ , then node  $j$  is idle during time slot  $t$ .

If  $W_{j,t+T} = W_{j,t} \forall j, t$ , then the schedule is periodic with period  $T$ . It can be written as a matrix of order  $N \times T$  denoted by  $\mathbf{W}^{(T)}$  and

$$W_{j,t} = W_{j,(t \bmod T)}^{(T)}. \quad (2.5)$$

#### **Necessary condition for transmission**

Node  $j$  transmits a message to node  $i$  during time slot  $t$  only if node  $i$  is able to successfully receive the message during time slot  $t + D_{ij}$ , i.e.,

$$W_{j,t} = i \Leftrightarrow W_{i,t+D_{ij}} = -j \quad \forall i \neq j. \quad (2.6)$$

#### **Necessary condition for successful reception**

To ensure successful reception at time slot  $t$  of a transmitted message from node  $j$ , it is required that no other nodes transmit messages that arrive at node  $i$  during time slot  $t$ . Therefore,

$$W_{i,t} = -j \Rightarrow W_{k,t-D_{ik}} \leq 0 \quad k \neq i. \quad (2.7)$$

TABLE 2.1: SCHEDULE FOR AN EQUILATERAL TRIANGLE NETWORK

	Time Slot 1	Time Slot 2	Time Slot 3	Time Slot 4
Node 1	TX to 2	TX to 3	RX from 3	RX from 2
Node 2	RX from 3	RX from 1	TX to 1	TX to 3
Node 3	RX from 2	TX to 1	RX from 1	TX to 2

### Throughput

The average throughput  $S$  of a schedule with period  $T$  can be computed by counting the number of receptions in the schedule  $\mathbf{W}^{(T)}$ .

$$S = \frac{1}{T} \sum_t \sum_j \mathbb{1}(W_{j,t}^{(T)} < 0) \quad (2.8)$$

where  $\mathbb{1}(E)$  is the indicator function of an event  $E$ , with value of 1 if  $E$  is true and 0 otherwise. In the case where the period of the schedule computed is not known, the approximate throughput is computed by counting the number of receptions over a large number of time slots  $T'$ . In that case, the approximate throughput  $S'$ , computed over  $T'$  slots is given by:

$$S' = \frac{1}{T'} \sum_{t=1}^{T'} \sum_{j=1}^N \mathbb{1}(W_{j,t} < 0). \quad (2.9)$$

The throughput defined in (2.8) & (2.9) only count the number of receptions but do not take into account the utilization of the time slots.  $\rho$ -throughput denoted by  $S_\rho$ , is defined as:

$$S_\rho = S(1 - \rho^+ - \rho^-) \quad (2.10)$$

and takes into account the time slot utilization [1].

### Example delay matrix & schedule

The delay matrix and schedule matrix for a three node equilateral triangle are as follows:

$$\mathbf{D} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \mathbf{W}^{(4)} = \begin{bmatrix} 2 & 3 & -3 & -2 \\ -3 & -1 & 1 & 3 \\ -2 & 1 & -1 & 2 \end{bmatrix}. \quad (2.11)$$

The above-mentioned delay matrix represents a network geometry where the nodes are placed such that they make an equilateral triangle with the link propagation delays as one unit of time slot length. The throughput optimal schedule for the considered network is shown in Table 2.1. The schedule can be interpreted as follows: In first time slot, node 1 transmits a message to node 2, and in the second time slot, node 2 receives a message from node 1. Both nodes 1 and 3 transmit to each other in times slot 2 and receive the messages in time slot 3. Note that the interfering messages from the transmissions in time slot 2 overlap at time slot 3 at node 2 and hence this slot is utilized as transmission slot with node 2 transmitting to node 1. Rest of the schedule can be interpreted in a similar way. Also, note that the period of the schedule in this example is  $T = 4$  and the schedule repeats itself for every 4 time slots. The equivalent schedule matrix is shown in (2.11). The above schedule example is taken from [1] for illustration.

## Chapter 3

### Slotted Schedules with Transmission Power Control

---

We presented in Chapter 2, the recent studies which exploit large propagation delays to allow concurrent transmissions in the network. Given a suitable network geometry, it has been shown in [1] that the network throughput increases linearly with the number of nodes, provided optimal transmission schedules are adopted.

In this chapter, we take advantage of the results from [1] and investigate the effect of limiting the interference range by controlling the transmission power on network throughput. In [1], the authors consider networks where all the nodes lie in the interference range of any transmission. In fully-connected networks where the upper bound can be achieved, limiting the interference range through transmission power control would only result in reducing the energy consumption. However, for many realistic networks with arbitrary geometries, the upper bound on throughput may not be achieved. In such cases, limiting the interference range by controlling the transmission power will cause the network throughput to increase significantly.

We consider large number of random network geometries and show through simulations that, by limiting the interference range, we can design transmission schedules with significant improvement in throughput. We also present some

examples of network geometries where the upper bound  $\frac{N}{2}$  is achieved due to power control, but not otherwise. The work presented in this chapter is published in [86].

### 3.1 System Model and Assumptions

The system model and assumptions are as follows:

1. We consider an  $N$  node network deployed randomly in a 2D space of  $H \times H$  meters. Let  $\mathbf{x}_j$  be the position vector of node  $j$  in the wireless network deployed such that,

$$\mathbf{x}_j = \begin{bmatrix} p \\ q \end{bmatrix}, \quad p, q \sim U(0, H) \quad (3.1)$$

where  $p, q$  are uniformly distributed between 0 and  $H$ .

2. The nodes in the network are assumed to be half-duplex in nature, i.e., a node cannot receive and transmit simultaneously. All the transmissions are assumed to be unicast and intended to their corresponding destinations, i.e., a node cannot receive from or transmit to more than one node simultaneously.
3. We adopt a protocol channel model [87] and denote by  $\alpha$ , the ratio of interference range to the communication range. These ranges are function of the transmission power  $P$ . Let  $R_i$  denote the interference range and  $R_c$ , the communication range in the acoustic channel for a given transmission

power  $P$ . We can express  $\alpha$  as:

$$\alpha = \frac{R_i}{R_c} = \frac{\left(\frac{R_i}{c\tau}\right)}{\left(\frac{R_c}{c\tau}\right)} = \frac{D_i}{D_c} \quad (3.2)$$

where  $D_i$  and  $D_c$  are the propagation delays corresponding to the interference and communication range respectively in the acoustic medium, in units of time slot length  $\tau$ . If the transmission power is set such that the communication range is greater than or equal to the size of the network considered, any transmission on a particular node is heard by all other nodes in the network resulting in the single collision domain. Although, note that in multi-hop networks, the communication range, and the corresponding interference range will not include all the nodes in the network and might result in multiple partially overlapping collision domains. In wireless radio networks, the interference range is often considered to be approximately twice the communication range [88], [89]. We assume the same factor in UWA network and use  $\alpha = 2$  in our simulations.

4. The propagation delay corresponding to the maximum interference range in the network considered is denoted by  $G$  and is given by:

$$G = \max_{i,j} \alpha D_{ij} \quad (3.3)$$

where  $i, j \in \{1, \dots, N\}$ .

## 3.2 Scheduling Problem

Before formulating the scheduling problem, we present a method for selecting the time slot length to compute the closest integer delay matrix representing the random network geometry generated. This delay matrix is used to compute the schedules, with and without power control.

### 3.2.1 Choosing time slot length

We know from [1] that the utilization efficiency of a time slot of length  $\tau$  is given by:

$$\eta = \frac{\tau(1 - \rho^+ - \rho^-)}{\tau} = 1 - \rho^+ - \rho^-. \quad (3.4)$$

In reality, the values of time slot length  $\tau$  are constrained to those allowed by the underwater acoustic modems. To be more precise, the packet lengths are constrained by the modem configuration and capability. These constraints translate to restrictions on time slot lengths in order to efficiently utilize the slots. We denote the minimum and maximum possible time slot lengths that can be set by  $\tau_{\min}$  and  $\tau_{\max}$  respectively.

A simple one dimensional optimization problem needs to be solved in order to select the time slot length  $\tau$  which would provide the closest integer delay matrix yielding maximum utilization of time slots. We use a brute-force method to find an optimal time slot length. The problem is formally written as follows:

$$\begin{aligned} \min_{\tau} \quad & \rho^+ + \rho^- \\ \text{s. t.} \quad & \tau = \{\tau_{\min} + i\Delta x ; i \in \mathbb{Z}^+ \cap [1, \lfloor \frac{\tau_{\max} - \tau_{\min}}{\Delta x} \rfloor]\}. \end{aligned} \quad (3.5)$$

where  $\Delta x$  is the smallest incremental step size in which the time slot length can be altered in the modem. We use  $\Delta x = 1$  ms in our simulations, but it may be varied to model different modems. For each value of time slot length between  $\tau_{\min}$  and  $\tau_{\max}$ , the delay matrices  $\mathbf{D}$  and  $\mathbf{D}'$  are computed using (2.1) and (2.2) and the corresponding values of  $\rho^+$  and  $\rho^-$  are computed using (2.3) and (2.4) further to find the value of the objective function in the above optimization problem. The optimal time slot length  $\tau^*$  found is then used to compute the delay matrix  $\mathbf{D}$  as given below:

$$D_{ij} = \frac{|\mathbf{x}_i - \mathbf{x}_j|}{c\tau^*}. \quad (3.6)$$

The delay matrix  $\mathbf{D}$  computed is then rounded off to the closest integer delay matrix  $\mathbf{D}'$  as shown in (2.2).

### **3.2.2 Link scheduling optimization problem**

The scheduling problem is formulated as a sequential decision problem (SDP) [1], where at each time slot  $t$ , the decisions are taken on which nodes should transmit to which other node in the network and the schedule is updated accordingly. A deterministic SDP is defined by the state space, action space, the transition function  $f$  which describes how the state changes as a result of the actions, and the reward function, which evaluates the immediate performance of the action taken.

### State space

The state of the decision problem is denoted by  $\mathbf{W}^{\{t,u\}}$ , which represents the partial schedule containing all the transmissions between time slots  $t$  and  $t - \alpha G$  and  $u - 1$  transmission decisions already taken in time slot  $t$ . The propagation delay corresponding to the maximum transmission range in the network is denoted by  $G$ , and hence  $\alpha G$  is the propagation delay corresponding to maximum interference range in the network. The transmission on this link would affect  $\alpha G$  slots in future. Therefore, in order to take decision at time slot  $t$ , it is enough to consider transmissions which occurred in the past till  $t - \alpha G$  slots. All possible partial matrices that  $\mathbf{W}^{\{t,u\}}$  can take, form the state space of the decision problem and is denoted by  $\mathcal{W}$ .

### Action space

The  $u^{\text{th}}$  action to be taken in time slot  $t$  is denoted by  $\mathbf{x}^{\{t,u\}}$ . The  $u^{\text{th}}$  action  $\mathbf{x}^{\{t,u\}}$  taken is a tuple  $(j, k)$ , which denotes node  $j$  transmitting to node  $k$ . Let  $M_t$  be the total number of transmissions during time slot  $t$ . Since we have  $N$  nodes,  $M_t \leq N$ . The action space from which the  $u^{\text{th}}$  action  $\mathbf{x}^{\{t,u\}}$  is chosen at time slot  $t$  is denoted by  $\mathcal{X}^{\{t,u\}}$ .

### Transition function

As a result of the  $u^{\text{th}}$  transmission decision taken in time slot  $t$  in the state  $\mathbf{W}^{\{t,u\}}$ , the state changes according to the transition function  $f : \mathcal{W} \times \mathcal{X}^{\{t,u\}} \rightarrow \mathcal{W}$  :

$$\mathbf{W}^{\{t,u+1\}} = f(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}) \quad \forall u < M_t \quad (3.7)$$

$$\mathbf{W}^{\{t+1,1\}} = f(\mathbf{W}^{\{t,M_t\}}, \mathbf{x}^{\{t,M_t\}}). \quad (3.8)$$

### Reward function

Due to the action taken when in a particular state, an immediate scalar reward denoted by  $r(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}})$  is obtained according to the reward function  $r : \mathcal{W} \times \mathcal{X}^{\{t,u\}} \rightarrow \mathbb{R}$ . Note that the reward due to any action is the number of successful transmissions added to the schedule due to that action. Since every action  $(j, k)$  is just one transmission, the reward is 1 every time an action is taken. The reward evaluates the immediate effect of the transition from one state to another but does not say anything about its long-term effect.

### Policy

The actions are chosen according to a policy  $\pi : \mathcal{W} \rightarrow \mathcal{X}^{\{t,u\}}$ , using:

$$\mathbf{x}^{\{t,u\}} = \pi(\mathbf{W}^{\{t,u\}}). \quad (3.9)$$

Usually, the goal is to find an optimal policy that maximizes the return, starting from any initial state. The return is the sum of rewards along a trajectory starting at some initial state. It represents the reward obtained due to the sequence of decisions taken in the long run.

### Q-Value function

A way to characterize policies is by using their value functions. Two types of value functions exist – state value functions (V-functions) and state-action value functions (Q-functions) [90]. The Q-function  $Q^\pi : \mathcal{W} \times \mathcal{X}^{\{t,u\}} \rightarrow \mathbb{R}$  of a policy  $\pi$  gives the return obtained when starting from a given state, applying a given

action, and following policy  $\pi$  thereafter. The optimal Q-function is defined as the best Q-function that can be obtained by any policy:

$$Q^*(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}) = \max_{\pi} Q^{\pi}(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}). \quad (3.10)$$

Any policy  $\pi^*$  that selects at each state, an action with the largest optimal Q-value, i.e., that satisfies:

$$\pi^*(\mathbf{W}^{\{t,u\}}) \in \operatorname{argmax}_{\mathbf{x}^{\{t,u\}}} Q^{\pi}(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}) \quad (3.11)$$

is optimal. In general, for a given Q-function, a policy  $\pi$  that satisfies:

$$\pi(\mathbf{W}^{\{t,u\}}) \in \operatorname{argmax}_{\mathbf{x}^{\{t,u\}}} Q(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}) \quad (3.12)$$

is said to be greedy in  $Q$ . So finding an optimal policy can be done by first finding  $Q^*$ , and then using (3.11) to compute a greedy policy in  $Q^*$  [90].

The Q-functions  $Q^{\pi}$  and  $Q^*$  are recursively characterized by the Bellman equations: [90], [91]:

$$Q^*(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}) = r(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}) + \max_{\mathbf{x}^{\{t,u+1\}}} Q(f(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}), \mathbf{x}^{\{t,u+1\}}). \quad (3.13)$$

Usually, the second term above is preceded by a discounting factor if the infinite horizon time is considered in computing Q-Value function. The discounting ensures that the return will always be bounded if the rewards are bounded. However, since we look at a finite time horizon in future as will be explained in the next section, we do not consider discounting in this formulation. The reward

for a single transmission decision is always 1, and hence it does not affect the decision making, therefore,

$$Q^*(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}) = \max_{\mathbf{x}^{\{t,u+1\}}} Q(f(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}), \mathbf{x}^{\{t,u+1\}}). \quad (3.14)$$

Note that if we have optimal Q-values for each state-action pair possible, we can take decisions optimally in every state. Hence, finding an optimal Q-value is of great interest to us. Solving the above problem shown in (3.13) using traditional methods like Q-value iteration is computationally intensive due to the large state space and action space [90], [91]. Instead, we follow a similar approach used in [1] where an approximate value function is defined for each state. If we denote the approximate Q-value function for each state-action pair by  $Q^\dagger(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}})$ , then the optimal decision is taken as follows:

$$\mathbf{x}^{\{t,u\}*} = \arg \max_{\mathbf{x}^{\{t,u\}}} Q^\dagger(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}). \quad (3.15)$$

### Approximate Q-Value function

At each state, based on the particular action  $(j, k)$  chosen, the partial schedule is updated and lands in a different state. An approximate Q-Value function thus needs to be defined for the state-action pair  $Q^\dagger(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}})$ . The Q-Value function should capture the ability of accommodating future potential transmissions when it is in the current state and a particular action is taken. As we want throughput to be maximized, the decision which supports maximum number of future potential transmissions is considered a good decision. The

approximate Q-Value function is written as:

$$Q^\dagger(\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}) = \sum_{j=1}^N \sum_{k=1}^N \sum_{\zeta=0}^{\alpha G} Z_{jk\zeta}(f((\mathbf{W}^{\{t,u\}}, \mathbf{x}^{\{t,u\}}))) \quad (3.16)$$

where  $Z_{jk\zeta}(\mathbf{W}^{\{t,u\}})$  is a *transmission indicator function* with value 1, if a transmission from node  $j$  to node  $k$  is allowed between time slots  $t$  and  $t + \alpha G$ , and 0 otherwise, given the partial schedule  $\mathbf{W}^{\{t,u\}}$ .

### Transmission indicator function

We list here the feasibility constraints under which the transmission from node  $j$  to node  $k$  is allowed or disallowed at time slot  $t + \zeta$ , where  $\zeta$  ranges from 0 to  $\alpha G$ , to capture the potential of accommodating future transmissions till  $\alpha G$  slots ahead from time slot  $t$ .

- a) Self transmissions are not allowed, i.e., a node is not permitted to transmit a message to itself.

$$Z_{jk\zeta}(\mathbf{W}^{\{t,u\}}) = 0 \text{ if } j = k. \quad (3.17)$$

- b) If node  $j$  in time slot  $t + \zeta$  is already scheduled to transmit or receive a message from some transmission that occurred in earlier time slot then node  $j$  is not permitted to transmit a message to node  $k$  and hence

$$Z_{jk\zeta}(\mathbf{W}^{\{t,u\}}) = 0 \text{ if } W_{j,t+\zeta}^{\{t,u\}} \neq 0. \quad (3.18)$$

- c) If node  $k$  is already receiving a message in time slot  $t + \zeta + D'_{jk}$  from some other node in the network, then the transmission from node  $j$  to node  $k$

should not be permitted in time slot  $t + \zeta$ , which implies

$$Z_{jk\zeta}(\mathbf{W}^{\{t,u\}}) = 0 \text{ if } W_{k,t+\zeta+D'_{jk}}^{\{t,u\}} \neq 0. \quad (3.19)$$

- d) If there exists a node  $i$  which has transmitted in such a slot  $t+\zeta+D'_{jk}-D'_{ik}$ , that its reception, even though not intended at node  $k$ , interferes with the transmission from node  $j$  to node  $k$  then node  $j$  must not be permitted to transmit to node  $k$ . This can be written as

$$Z_{jk\zeta}(\mathbf{W}^{\{t,u\}}) = 0 \text{ if } \exists i \text{ s.t. } W_{i,t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}} > 0. \quad (3.20)$$

- e) If there exists a link from node  $l$  to node  $i$  in the network such that the transmission from node  $l$  and its reception at node  $i$  happen in such a slot that the transmission from node  $j$  to node  $k$  would be interfering at the node  $i$ , then node  $j$  must not be permitted to transmit.

$$Z_{jk\zeta}(\mathbf{W}^{\{t,u\}}) = 0 \text{ if } \exists l, i \text{ s.t. } W_{l,t+\zeta+D'_{ji}-D'_{li}}^{\{t,u\}} = i. \quad (3.21)$$

- f) Finally if none of the above constraints are satisfied, then node  $j$  should be permitted to transmit to node  $k$  in time slot  $t + \zeta$  and hence

$$Z_{jk\zeta}(\mathbf{W}^{\{t,u\}}) = 1. \quad (3.22)$$

### Limiting interference range by power control

Now we shall consider controlling the transmission power in order to limit the interference range, and revisit the above-listed scheduling feasibility constraints. We assume that minimum transmission power  $P_{jk}$ , is used to transmit a message from node  $j$  to node  $k$ . If  $\alpha$  is the ratio of interference range to communication range, then by transmitting at minimum power between node  $j$  to node  $k$ , the message would be heard by all the nodes in the interference range of node  $j$  which is  $\alpha(|\mathbf{x}_j - \mathbf{x}_k|)$ . This implies that the message would be heard till  $\alpha D_{jk}$  time slots in future starting from the time slot in which node  $j$  transmitted. Note that the interference constraint due to  $\alpha$  should be applied to the non-integer version of the delay matrix  $\mathbf{D}$  instead of the integer delay matrix  $\mathbf{D}'$ . The reason is that, even if some fraction of the time slots get affected due to the interference, they become unavailable to be used for reception.

We note that the constraints (3.17),(3.18) and (3.19) remain the same even if we limit the interference range. Consider the constraint (3.20) and assume that while searching for node  $i$  we find such a node in the network satisfying the constraint (3.20). This implies <sup>1</sup>  $W_{i,t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}} > 0$ . This also implies that node  $i$  is transmitting to node  $W_{i,t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}}$  in time slot  $t + \zeta + D'_{jk} - D'_{ik}$ . Since the transmission link found is between node  $i$  and node  $W_{i,t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}}$ , the following must be true if node  $k$  lies in the interference range of transmission

---

<sup>1</sup> $\mathbf{W}^{\{t,u\}}$  denotes the state as well as the partially filled schedule matrix. Therefore,  $W_{j,t}$  denotes the element at the  $t^{\text{th}}$  column and  $j^{\text{th}}$  row of the partially filled matrix  $\mathbf{W}^{\{t,u\}}$  and hence these elements denote the node address when they are positive.

from node  $i$ :

$$\begin{aligned}
 t + \zeta + D'_{jk} - D'_{ik} + \lceil \alpha D_{i, W_{i, t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}}} \rceil &\geq t + \zeta + D'_{jk} \\
 \Rightarrow D'_{ik} &\leq \lceil \alpha D_{i, W_{i, t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}}} \rceil.
 \end{aligned} \tag{3.23}$$

Therefore, the constraint (3.20) should be now modified as follows:

$$Z_{jk\zeta}(\mathbf{W}^{\{t,u\}}) = 0 \text{ if } \exists i \text{ s.t.}$$

$$W_{i, t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}} > 0 \ \& \ D'_{ik} \leq \lceil \alpha D_{i, W_{i, t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}}} \rceil. \tag{3.24}$$

Now we consider constraint (3.21) and examine it in the case of limiting the interference range. The condition must include not only that there exists such nodes  $l$  and  $i$ , but also that node  $i$  must lie in the interference range of node  $j$ 's transmission to node  $k$ , i.e.,

$$D'_{ji} \leq \lceil \alpha D_{jk} \rceil. \tag{3.25}$$

Therefore, constraint (3.21) should now be modified as follows:

$$Z_{jk\zeta}(\mathbf{W}^{\{t,u\}}) = 0 \text{ if } \exists l, i \text{ s.t.}$$

$$W_{l, t+\zeta+D'_{ji}-D'_{li}}^{\{t,u\}} = i \ \& \ D'_{ji} \leq \lceil \alpha D_{jk} \rceil. \tag{3.26}$$

To put it all together, the transmission indicator function including power control

is summarized as follows:

$$Z_{jk\zeta}(\mathbf{W}^{\{t,u\}}) = \begin{cases} 0 & \text{if } j = k \\ 0 & \text{if } W_{j,t+\zeta}^{\{t,u\}} \neq 0 \\ 0 & \text{if } W_{k,t+\zeta+D'_{jk}}^{\{t,u\}} \neq 0 \\ 0 & \text{if } \exists i \text{ s.t. } W_{i,t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}} > 0 \ \& \\ & D'_{ik} \leq \lceil \alpha D_{i,W_{i,t+\zeta+D'_{jk}-D'_{ik}}^{\{t,u\}}} \rceil \\ 0 & \text{if } \exists l, i \text{ s.t. } W_{l,t+\zeta+D'_{ji}-D'_{li}}^{\{t,u\}} = i \ \& \\ & D'_{ji} \leq \lceil \alpha D_{jk} \rceil \\ 1, & \text{otherwise.} \end{cases} \quad (3.27)$$

#### Algorithm to take transmission decisions

The optimal time slot length  $\tau^*$  to be used is computed by solving the optimization problem as presented in the Section 3.2.1. We can compute the corresponding closest integer delay matrix as follows :

$$D'_{ij} = \left\lceil \frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{c\tau^*} \right\rceil. \quad (3.28)$$

The Algorithm 1 summarizes the procedure to take decisions at each time slot.

### 3.3 Performance Evaluation

We evaluate and compare the performance of the proposed idea of limiting the interference range through power control using simulations. We compare

---

**Algorithm 1** Algorithm to take  $u^{\text{th}}$  transmission decision at time slot  $t$  with power control

---

**Require:**  $\mathbf{D}'$ ,  $\mathbf{D}$ , current state of the schedule  $\mathbf{W}^{\{t,u\}}$ ,  
 $\mathbf{W} \leftarrow \mathbf{W}^{\{t,u\}}$   
**while** true **do**  
    Compute  $Z$  from  $\mathbf{W}$  using (3.27)  
     $\mathcal{X} \leftarrow \{(j, k), \forall j, k \text{ s.t. } Z_{jk0} = 1\}$   
    **if**  $\mathcal{X}$  is empty **then**  
        **return**  $\mathbf{W}^{\{t+1,1\}} \leftarrow \mathbf{W}, M_t \leftarrow u$   
    **end if**  
    Compute  $Q^\dagger(\mathbf{W}, \mathbf{x})$  using (3.16)  
     $\mathbf{x}^* = \operatorname{argmax} Q^\dagger(\mathbf{W}, \mathbf{x})$   
     $\mathbf{W} = f(\mathbf{W}, \mathbf{x}^*)$   
**end while**

---

throughput, computed when the schedules are found, with and without power control. We also look at different network geometries found through simulations, for which the upper bound on throughput is achieved only due to power control. We consider 200 randomly generated  $N$  node networks in a 2D space of  $H \times H$  meters, where the number of nodes  $N$  considered is in the range from 2 to 8 nodes. We set  $H = 3000$  m,  $\tau_{\min} = 45$  ms,  $\tau_{\max} = 3000$  ms,  $\Delta x = 1$  ms and  $\alpha = 2$ . The closest integer delay matrix is computed for each randomly generated network geometry, after selecting an appropriate time slot length  $\tau$ , by solving the optimization problem presented in Section 3.2.1. Algorithm 1 is used to compute the schedules with power control. Note that in Algorithm 1, while computing transmission indicator function  $Z(\cdot)$ , if we do not limit the interference range, i.e., we use the constraints (3.20) & (3.21) instead of constraints (3.24) & (3.26), then we compute schedules without applying power control.

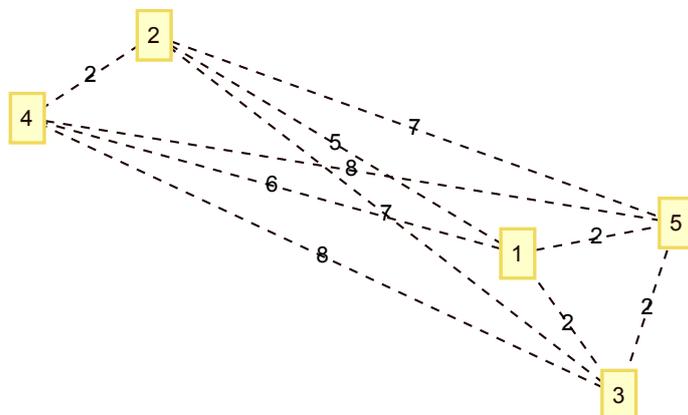


Figure 3.1: 5 node trivial random network geometry with throughput  $S = 2.5$  and  $\rho$ -throughput  $S_\rho = 1.99$ .

### 3.3.1 Geometries found with throughput gain

Some of the network geometries found are shown in Fig. 3.1, 3.2 & 3.3. Each network geometry plotted is on a different scale and is only representative of the shape of the network. Also note that the edges are labeled with approximated values of the propagation delays between nodes, which are taken from the corresponding integer delay matrix of the network geometry.

#### Trivial geometries with $\frac{N}{2}$ throughput

Consider a network geometry with the following delay matrix which was found while 5 node random network geometries were generated (see Fig. 3.1) and an appropriate time slot length  $\tau = 168$  ms was chosen to approximate it to the closest integer delay matrix. The delay matrix  $\mathbf{D}$  computed is given below:

$$\mathbf{D} = \begin{bmatrix} 0 & 5.26 & 2.16 & 6.30 & 2.00 \\ 5.26 & 0 & 7.30 & 1.88 & 6.88 \\ 2.16 & 7.30 & 0 & 8.11 & 2.26 \\ 6.30 & 1.88 & 8.11 & 0 & 8.15 \\ 2.00 & 6.88 & 2.26 & 8.15 & 0 \end{bmatrix}. \quad (3.29)$$

The closest integer delay matrix, when used with the scheduling algorithm with power control, generates a schedule with period  $T = 8$ , which is shown below:

$$\mathbf{W}^{(8)} = \begin{bmatrix} 5 & 5 & -5 & -5 & -3 & -3 & 3 & 3 \\ 4 & 4 & -4 & -4 & 4 & 4 & -4 & -4 \\ -1 & -1 & 1 & 1 & 5 & 5 & -5 & -5 \\ 2 & 2 & -2 & -2 & 2 & 2 & -2 & -2 \\ 1 & 1 & -1 & -1 & 3 & 3 & -3 & -3 \end{bmatrix}. \quad (3.30)$$

Although counting the number of successful receptions in one period of the above schedule would result in a throughput of 2.5, we need to be mindful of the approximations in propagation delays made while computing schedules. However, we can see that due to power control, two sub-networks are formed with links involving nodes 2, 4 and nodes 1, 3, 5 forming an equilateral triangle. The  $\rho$ -throughput can be computed by adding the  $\rho$ -throughput of each sub-network as given below:

$$S_\rho = 1(1 - 0.12) + 1.5(1 - 0 - 0.26) = 1.99.$$

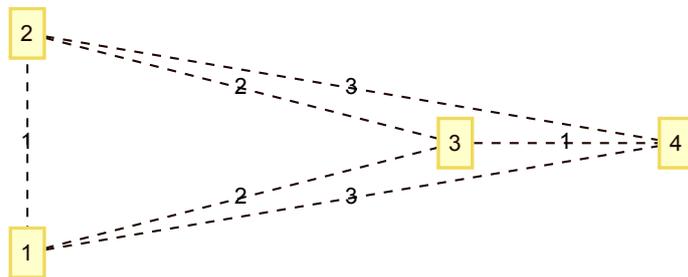


Figure 3.2: 4 node non-trivial network geometry with throughput  $S = 2$  and  $\rho$ -throughput  $S_\rho = 2$ .

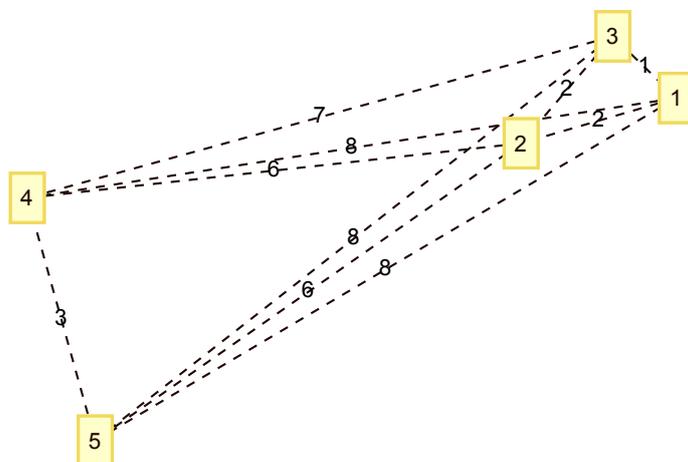


Figure 3.3: 5 node non-trivial random network geometry with throughput  $S = 2.5$  and  $\rho$ -throughput  $S_\rho = 1.45$ .

Also note that the links which are used in the schedule are those which cause minimum interference.

### Non-Trivial geometries with $\frac{N}{2}$ throughput

The above example of network geometry do not give insight into how power control results in schedules with higher throughput, when we cannot spatially separate the network into smaller sub-networks. Consider a 4 node network

geometry (see Fig. 3.2) with the following delay matrix:

$$\mathbf{D} = \begin{bmatrix} 0 & 1 & 2 & 2.9788 \\ 1 & 0 & 2 & 2.9788 \\ 2 & 2 & 0 & 1 \\ 2.9788 & 2.9788 & 1 & 0 \end{bmatrix}. \quad (3.31)$$

For this network geometry, the schedule computed without power control results in a schedule with period  $T = 7$  as shown below:

$$\mathbf{W}^{(7)} = \begin{bmatrix} -4 & 3 & -4 & 2 & -2 & 2 & -2 \\ 3 & 0 & 3 & 1 & -1 & 1 & -1 \\ 0 & -4 & -2 & -1 & -2 & 0 & -4 \\ 3 & 0 & 0 & 0 & 1 & 3 & 1 \end{bmatrix}. \quad (3.32)$$

The above schedule yields throughput,  $S = \frac{11}{7} = 1.56$ . However, if the schedule is computed after limiting the interference range, we can achieve a throughput  $S = 2$  as shown below:

$$\mathbf{W}^{(2)} = \begin{bmatrix} 2 & -2 \\ 1 & -1 \\ 4 & -4 \\ 3 & -3 \end{bmatrix}. \quad (3.33)$$

We note that node 3 is in the interference range of transmission on links (1,2) & (2,1) and also nodes 1 & 2 are in the interference range of the link (3,4). Controlling the transmission power at nodes 1 & 2, results in no interference at node 4 and hence we can design the transmission schedule

which can achieve a throughput of 2 as shown above. The  $\rho$ -throughput of the above network is also 2. Note that the set of interfering links for this network, when used with the schedule computed with power control, is  $\mathcal{I} := \{(1, 2), (2, 1), (2, 3), (3, 2), (1, 3), (3, 1), (3, 4), (4, 3)\}$ . Hence while computing  $\rho^+$  and  $\rho^-$  we can ignore the approximations in the propagation delay of the links  $(2, 4)$  and  $(1, 4)$ . This results in the values of  $\rho^+$  and  $\rho^-$  to be 0. Consider another example, a 5 node network geometry as shown in Fig. 3.3 with the following delay matrix when used with time slot length  $\tau = 241$  ms:

$$\mathbf{D} = \begin{bmatrix} 0 & 1.88 & 1.03 & 7.63 & 7.85 \\ 1.88 & 0 & 1.64 & 5.77 & 6.05 \\ 1.03 & 1.64 & 0 & 7.06 & 7.65 \\ 7.63 & 5.77 & 7.06 & 0 & 2.95 \\ 7.85 & 6.05 & 7.65 & 2.95 & 0 \end{bmatrix}. \quad (3.34)$$

and the schedule is computed with a period of 12 and with throughput  $S = 2.5$  as shown below:

$$\mathbf{W}^{(12)} = \begin{bmatrix} 3 & 2 & -2 & 2 & -2 & -3 & -3 & -3 & 2 & -2 & 3 & 3 \\ 1 & 3 & 1 & -1 & -3 & -1 & 3 & 1 & 3 & -3 & -1 & -3 \\ -1 & -1 & 2 & -2 & 1 & 1 & 1 & 2 & -2 & 2 & -2 & -1 \\ 5 & 5 & 5 & -5 & -5 & -5 & 5 & 5 & 5 & -5 & -5 & -5 \\ 4 & 4 & 4 & -4 & -4 & -4 & 4 & 4 & 4 & -4 & -4 & -4 \end{bmatrix}.$$

Similar to the example above, while computing  $\rho$ -throughput, the set

of interfering links is enumerated and the values of  $\rho^+$  and  $\rho^-$  are computed. It results in  $\rho^+ = 0.0549$  and  $\rho^- = 0.3627$ , and hence  $S_\rho = 2.5(1 - 0.0549 - 0.3627) = 1.456$ .

### 3.3.2 Simulation results

#### Percentage increase in average throughput

The average throughput over 200 random network geometries needs to be computed for two cases, i.e., with and without power control in order to compute % increase in average throughput. Fig. 3.4 shows the throughput values computed with 95% confidence intervals as a notched box plot. Since the notches in the box plot do not overlap, we can conclude with 95% confidence that the true medians do differ. Moreover, this justifies the choice of averaging over 200 random network geometries.

As we see in Fig. 3.5, the percentage increase in the average throughput computed is plotted as a function of the number of nodes in the network. If we denote the approximate throughput  $S'$ , computed over  $T'$  time slots without power control, by  $S'_{\text{wopc}}$ , and the throughput, computed with power control, by  $S'_{\text{wpc}}$ , then the percentage increase is computed by:

$$\beta = \frac{S'_{\text{wpc}} - S'_{\text{wopc}}}{S'_{\text{wopc}}} \times 100. \quad (3.35)$$

From the plot, it is clear that limiting the interference range results in improving the average throughput. For an 8 node network the percentage increase in average throughput is close to 15%. Note that without power control, in a

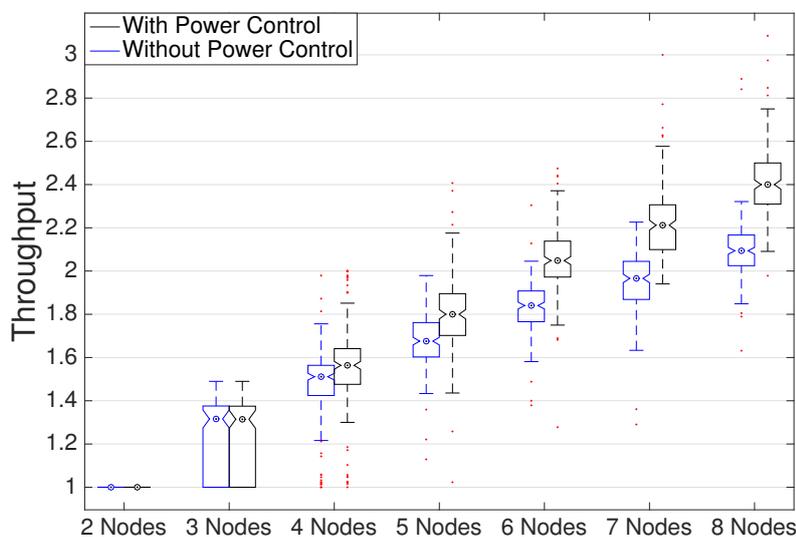


Figure 3.4: Throughput values showing 95% confidence interval is plotted as a function of the number of nodes in the network.

fully connected network, the number of transmission opportunities will reduce as compared to the case with power control as the number of nodes in the network increase. Since the links to be scheduled are not specified and the algorithm selects throughput-maximizing links, power control might result in the formation of many smaller fragmented sub-networks which are throughput-maximizing. This explains why the % increase in average throughput is seen to increase in Fig. 3.5 and the behavior will remain even when the number of nodes in the network is increased beyond 8 nodes.

Fig. 3.6 shows the average throughput and maximum throughput computed as a function of number of nodes. It is clearly seen that the throughput values computed with power control, are greater than or equal to the values computed without power control, and hence it is always better to limit the interference range.

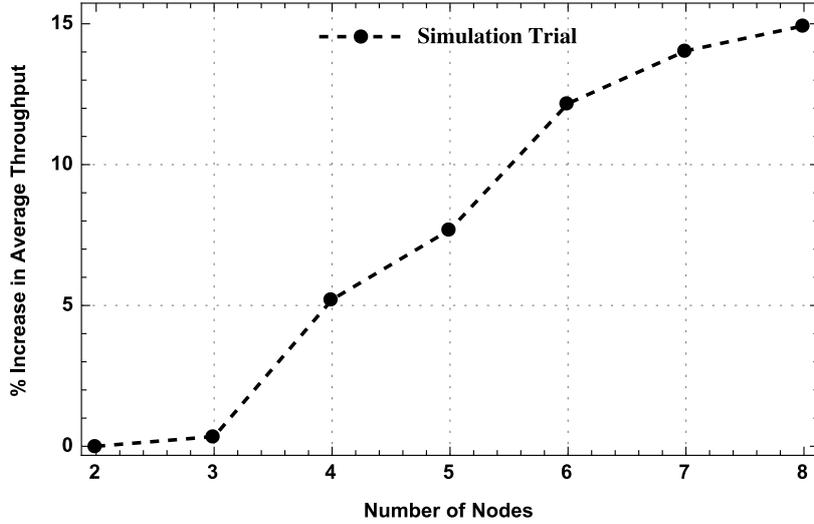


Figure 3.5: Percentage increase in average throughput due to power control.

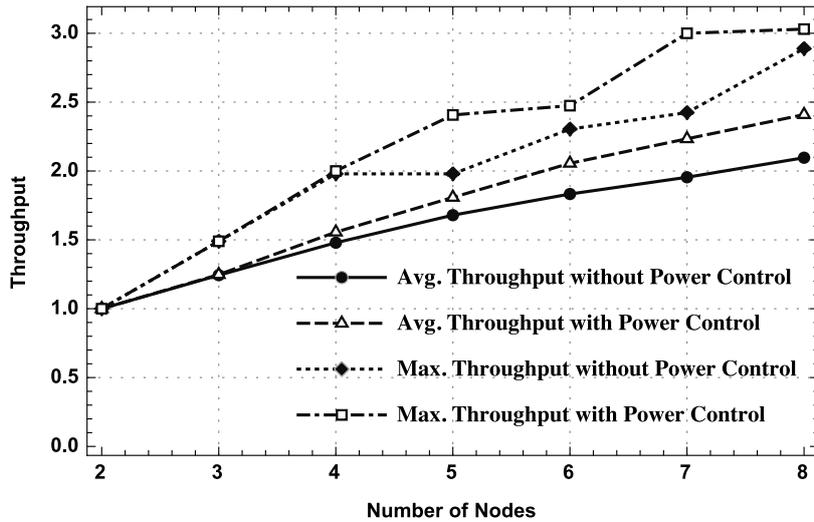


Figure 3.6: Average & Maximum throughput over 200 random network geometries.

### Histogram of throughput of random geometries

A histogram of throughput values computed for 5 node random network geometries is shown in Fig. 3.7. We can see that the maximum throughput values computed lie between 1.7 and 1.9 and hence it is no surprise to see that

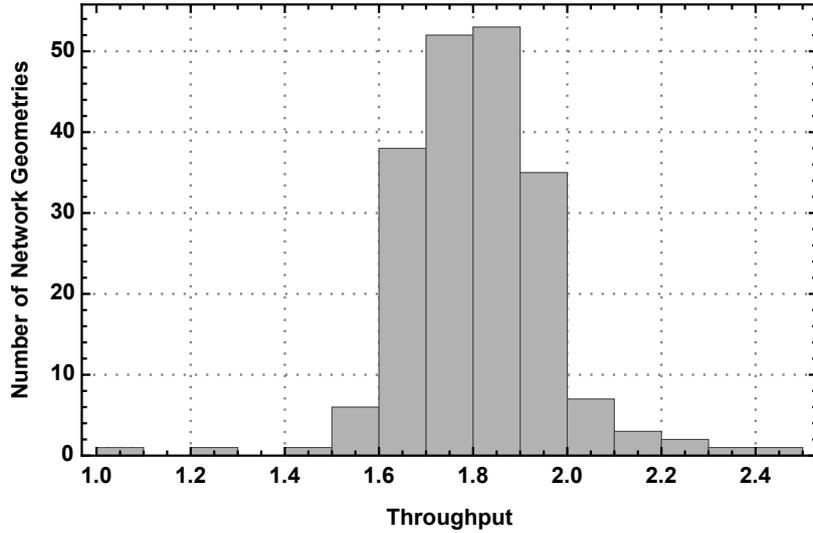


Figure 3.7: Histogram of throughput with power control for 200 randomly generated 5 node networks.

the average throughput value from Fig. 3.6 is in the same range for  $N = 5$ . We can also see how often the network geometries occur for which the upper bound is achieved. It is clear from Fig. 3.7, that there is a non-negligible probability of finding networks for which optimal throughput can be achieved.

### 3.4 Summary

We have shown through simulation that limiting the interference range by controlling the transmission power results in significant throughput increase. Limiting the interference range allows more transmission opportunities in the network and hence results in throughput that is closer to the upper bound  $\frac{N}{2}$ , when compared with the throughput of a fully-connected network. We have also shown instances of random network geometries for which the throughput upper bound is achieved due to power control, but not without. The network geometries presented demonstrate that power control along with exploitation

of propagation delay can result in schedules which yield high throughput. We also showed that the computation of  $\rho^+$  and  $\rho^-$  need not take into account the propagation delay between all links in the network. This results in better  $\rho$ -throughput than when power control is not applied.

## Chapter 4

### Unslotted Schedules with Variable Packet Lengths

---

In Chapter 3, we considered controlling transmission power to limit the interference while utilizing the propagation delays among the nodes and presented that more transmission opportunities are created resulting in much higher throughput for arbitrary network geometries. However, the solution is time-slotted and only those links are selected in the schedule which are throughput-maximizing. Time-slotted transmissions utilizing propagation delays are such that most of the interference overlaps with the transmission slots and the receiving slots are interference-free. However, the packet duration of each transmission is assumed to be fixed. A packet can be divided into several smaller packets of different size. The packet duration provides a degree of freedom that, if utilized, results in strategies that can be adopted to achieve throughput closer to the established upper bound.

In this chapter, we consider the unslotted transmission of packets with arbitrary size. Given the propagation delay between nodes in the network and the packet traffic demands on each link, we formulate an optimization problem for minimizing the fractional idle time in a frame (or period) of the schedule as a Mixed-Integer Linear Fractional Problem (MILFP). We compare our results to the recent advancements that exploit large propagation delays and result in

time-slotted and unslotted schedules with fixed packet duration. We also present schedules computed for various network geometries with arbitrary packet traffic demands. The work presented in this chapter is submitted for publication [92].

#### 4.1 System Model and Assumptions

The system model and assumptions are as follows:

1. A set  $\mathcal{L}$  of directed links is considered for scheduling in an UWA network, given the propagation delay between all nodes in the network and traffic demands on each link  $l \in \mathcal{L}$ . Each link  $l$  can be explicitly written as a 2-tuple  $(j, k)$  which represents a link where node  $j$  is the transmitter and node  $k$  is the receiver. The propagation delay corresponding to link  $(j, k)$  is  $D_{jk}$  in seconds. The packet traffic demand,  $\Lambda_{jk}$  for link  $(j, k)$ , is defined as the number of packets to be transmitted in a single frame on link  $(j, k)$ .
2. We allow having packets of different lengths for the same link. The time relative to the start of the frame, at which, node  $j$  starts transmitting the  $x^{\text{th}}$  packet to node  $k$  is  $t_{jk}^x$ , where  $x \in \{1, \dots, \Lambda_{jk}\}$ . The packet/transmission duration corresponding to the  $x^{\text{th}}$  transmission on link  $(j, k)$  is  $\tau_{jk}^x$ . The frame length of the schedule (also termed as period of the schedule in [1]) is  $T$ .
3. A central controller computes schedules based on the available information on the network topology and traffic demands while exploiting the large propagation delay between the nodes. The network considered is static (with small motion around the location at which nodes are deployed). The centralized optimal solution provides an upper bound on the performance

of any distributed or heuristic algorithm due to limited knowledge of the network topology in case of the former and dependence on the quality of heuristic in the case of latter. Such an upper bound allows evaluation of the performance of distributed and heuristic algorithms.

4. The mathematical formulation allows having packets of different lengths for the same link. We assume link traffic demands to be given in the case of the single-hop networks or are calculated by given end-to-end traffic demands in case of a multi-hop network with predetermined routing. The proposed framework may be extended to include routing in the optimization problem, but this is out of the scope of this work.
5. We consider a form of TDMA as in several recent works [1], [2], [85], [93]. We partition time into frames but do not further partition into slots as commonly done. This problem formulation allows us to have variable packet duration for each transmission with no slotting required. This plays an important role in achieving throughput gain. The objective considered for minimization is the fractional idle time in a frame. The idle time is the total time for which the nodes are neither transmitting nor receiving.
6. As already mentioned in Section 3.1, underwater acoustic modems are assumed to be half-duplex in nature, i.e., a node cannot receive and transmit simultaneously. All the transmissions are assumed to be unicast and intended to their corresponding destinations, i.e., a node cannot receive from or transmit to more than one node simultaneously.
7. We adopt a protocol channel model [87] and denote by  $\alpha$ , the ratio of

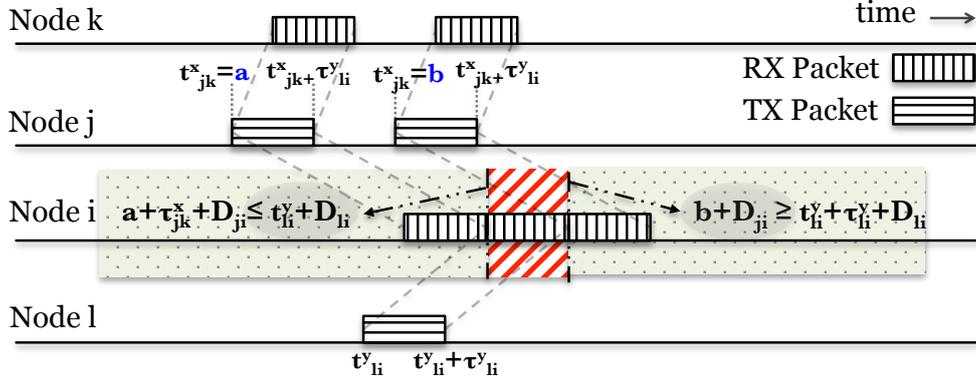


Figure 4.1: Illustration of the effect of propagation delay on transmission times of packets to be transmitted on links  $(j, k)$  and  $(l, i)$ .

interference range to the communication range as described in Section 3.1.

## 4.2 Problem Formulation

Consider a pair of links  $(j, k), (l, i) \in \mathcal{L}$  such that node  $j$  transmits its  $x^{\text{th}}$  packet to node  $k$  and node  $l$  transmits its  $y^{\text{th}}$  packet to node  $i$  (see Fig. 4.1). Fig. 4.1 illustrates the effect of propagation delay. The interference is caused at node  $i$ , only if, node  $j$  transmits at such a time  $t_{jk}^x$ , that it ends up in the hatched region (in red) marked in Fig. 4.1 at node  $i$ . The choice of  $t_{jk}^x \leq a$  or  $t_{jk}^x \geq b$  avoids the interference during reception time at node  $i$  and ends up in the dotted region (in green) instead. Note that the packet lengths  $\tau_{jk}^x$ 's and  $\tau_{li}^y$  are considered equal in Fig. 4.1 only for the purpose of illustration. For node  $j$ 's transmission to not interfere with node  $i$ 's reception, either of the following necessary conditions need to be satisfied:

$$t_{jk}^x + \tau_{jk}^x + D_{ji} \leq t_{li}^y + D_{li} \quad (4.1)$$

(or)

$$t_{jk}^x + D_{ji} \geq t_{li}^y + \tau_{li}^y + D_{li} \quad (4.2)$$

$\forall \{(j, k), (l, i) \in \mathcal{L} | D_{ji} \leq \alpha D_{jk}\}$ , where the condition  $D_{ji} \leq \alpha D_{jk}$  is satisfied when node  $i$  lies in the interference range of node  $j$ . The transmission start times  $t_{jk}^x, t_{li}^y$  and corresponding packet duration  $\tau_{jk}^x, \tau_{li}^y$  must be chosen such that the desired message at node  $i$  is interference-free. The conditions stated in (4.1) and (4.2) are illustrated in Fig. 4.1. Note that if the interference range is larger than the size of the network considered (single collision domain network), then each transmission in the network is heard by all the nodes. In this case the conditions in (4.1) and (4.2) applies to all the links in  $\mathcal{L}$ . We consider single collision domain network throughout, since this model allows us to study the effects of propagation delay for the worst case scenario, although the formulation is general enough to be used for both single-hop as well as multi-hop UWA networks. While presenting the scalability analysis in Section 4.6, we consider a multi-hop network with multiple partially overlapping collision domains.

If the transmission scheduling decisions are taken considering the propagation delay based constraints listed in (4.1) and (4.2), the corresponding receptions at the nodes will be interference-free, i.e., there will be no collisions within a frame. However, note that the following possibilities are not captured in the constraints listed in (4.1) and (4.2):

1. The transmissions from the previous frames interfering with reception in the current frame.
2. The transmission in the current frame interfering with possible reception in the subsequent frames.

Not considering the above conditions, will result in poor schedules. In order to consider the inter-frame constraints, we generalize the inequalities:

$$t_{jk}^x + \beta T + \tau_{jk}^x + D_{ji} \leq t_{li}^y + D_{li} \quad (4.3)$$

(or)

$$t_{jk}^x + \beta T + D_{ji} \geq t_{li}^y + \tau_{li}^y + D_{li} \quad (4.4)$$

$\forall \{(j, k), (l, i) \in \mathcal{L}\}$ , where  $\beta \in \mathbb{Z}$  is the integer constant which determines the number of adjacent frames in the past and future in time, that are considered. Note that setting  $\beta = 0$ , in (4.3) and (4.4) leads to (4.1) and (4.2) respectively.  $\beta = -1$ , corresponds to conditions considering one adjacent frame in the past while  $\beta = 1$ , corresponds to conditions considering one adjacent frame in the future.

There is no upper bound on the value of  $\beta$ , since the frame length can be arbitrarily small theoretically. In practice, the schedules with such smaller frame lengths are not implementable on underwater modems for practical use. In [1], the optimal schedules for the network geometries result in the frame lengths which are atleast greater than  $G$  (defined in (3.3)). With this reasonable assumption of limiting the value of frame length to be greater than the girth of the network,  $T > G$ , we limit our constraints to  $\beta = 0$ ,  $\beta = 1$  and  $\beta = -1$ . Consider the transmission of a packet in current frame at time  $t$  for duration  $\tau$ . Interference caused due to this transmitted packet in worst case only lasts till time  $t + G + \tau$ . Since we know  $G < T$ , we have  $t + G + \tau < t + T + \tau \Rightarrow t + G <$

$t + \beta T$ ,  $\beta = 1$ . In other words, in the worst case scenario, the interference is only limited within the next frame.

#### 4.2.1 Propagation delay constraints

Consider scheduling all links  $(j, k) \in \mathcal{L}$  in a frame. The number of packets to be transmitted on each link  $(j, k)$  is given by the traffic demand for that link  $\Lambda_{jk}$  packets/frame.  $x$  is the index associated with link  $(j, k)$  taking values  $1, \dots, \Lambda_{jk}$  and similarly  $y$  is the index for link  $(l, i)$  taking values  $1, \dots, \Lambda_{li}$  (e.g.  $t_{jk}^1, \dots, t_{jk}^{\Lambda_{jk}}, \tau_{jk}^1, \dots, \tau_{jk}^{\Lambda_{jk}}$  are variables associated with transmission time and packet duration for each link  $(j, k)$ ). Note that the conditions listed in (4.3) and (4.4) form a set of disjunctive constraints which results in the feasible set forming a non-convex region over which the search for the solution is required. There are two well-known methods for conversion of disjunctive constraints to conjunctive constraints: convex-hull reformulation [94] and Big-M reformulation [95], [96]. We use the Big-M transformation to convert the disjunctive constraints (4.3) & (4.4) into conjunction and rearrange them as following:

1.  $\beta = 0$  :

$$t_{jk}^x - t_{li}^y + \tau_{jk}^x \leq -(D_{ji} - D_{li}) + Mp_{jk,li}^{xy} \quad (4.5)$$

$$-t_{jk}^x + t_{li}^y + \tau_{li}^y \leq (D_{ji} - D_{li}) + M(1 - p_{jk,li}^{xy}) \quad (4.6)$$

2.  $\beta = 1$  :

$$t_{jk}^x - t_{li}^y + \tau_{jk}^x + T \leq -(D_{ji} - D_{li}) + Mq_{jk,li}^{xy} \quad (4.7)$$

$$-t_{jk}^x + t_{li}^y + \tau_{li}^y - T \leq (D_{ji} - D_{li}) + M(1 - q_{jk,li}^{xy}) \quad (4.8)$$

3.  $\beta = -1$  :

$$t_{jk}^x - t_{li}^y + \tau_{jk}^x - T \leq -(D_{ji} - D_{li}) + Mr_{jk,li}^{xy} \quad (4.9)$$

$$-t_{jk}^x + t_{li}^y + \tau_{li}^y + T \leq (D_{ji} - D_{li}) + M(1 - r_{jk,li}^{xy}) \quad (4.10)$$

where  $p_{jk,li}^{xy}$ ,  $q_{jk,li}^{xy}$  and  $r_{jk,li}^{xy}$  are the binary variables<sup>1</sup> associated with each pair of disjunctive constraints considered  $\forall (j, k), (l, i) \in \mathcal{L}$  and  $x \in \{1, \dots, \Lambda_{jk}\}, y \in \{1, \dots, \Lambda_{li}\}$ . The inequalities (4.3) and (4.4) are presented in (4.5), (4.6), (4.7), (4.8), (4.9) and (4.10) after the Big-M transformation for the values of  $\beta$  from the set  $\{-1, 0, 1\}$ .

#### 4.2.2 Duration between two consecutive transmissions

Since the packet traffic demands  $\Lambda_{jk}$  packets per frame for link  $(j, k)$ , the following constraint on transmissions from node  $j$  ensures the difference between two consecutive packet transmission times is atleast greater than the previous transmitted packet duration:

$$t_{jk}^{x+1} - t_{jk}^x \geq \tau_{jk}^x. \quad (4.11)$$

<sup>1</sup>With the binary variable taking value 0 or 1 (e.g.  $p_{jk,li} = 0$  or 1) along with a large enough value of parameter M, one of the constraints in the disjunctive pair becomes redundant. Note that smaller the value of M is, the tighter the Big-M reformulation can be. We select an arbitrarily large value of M for the transformation.

### 4.2.3 Allowing transmissions and receptions across frame boundary

A packet transmission in the current frame at time  $t_{jk}^x$ , in the worst case can cause interference till time  $t_{jk}^x + G + \tau_{jk}^x$ . To prevent the end of the packet transmission on the link causing maximum interference in future to cross the subsequent frame, we impose the following constraint:

$$t_{jk}^x + G + \tau_{jk}^x < 2T. \quad (4.12)$$

Note that the above constraint does not restrict the transmissions in the frame to be fully contained within the frame.

### 4.2.4 Throughput

The average throughput  $S$  of a schedule with frame length  $T$  can be computed by summing the total reception (or equivalently transmission) time on all the nodes in the network in one frame duration  $T$ :

$$S = \frac{1}{T} \sum_{j=1}^N \left[ \sum_{(k,j) \in \mathcal{L}} \sum_{x=1}^{\Lambda_{kj}} \tau_{kj}^x \right]. \quad (4.13)$$

### 4.2.5 Inclusion of packet headers

In practical modems, the packet consists of the header and payload. We can include the packet headers in the problem formulation by replacing  $\tau_{jk}^x$  with  $\overline{\tau}_{jk}^x$  in all constraints except the objective function, where

$$\overline{\tau}_{jk}^x = \mathbf{t}_p + \tau_{jk}^x \quad (4.14)$$

and  $\tau_p$  is the packet header duration whereas  $\tau_{jk}^x$  is the payload duration. We present the results including the non-negligible packet headers in Section 4.4.1.

#### 4.2.6 Objective function

The objective considered in the recent work [2] is to minimize the frame length  $T$ , and the packet duration is fixed. Even if the packet duration is considered a variable, minimizing  $T$ , results in packet duration which are equal in size for all transmissions to the lower bound set on the packet duration. We demonstrate this in Section 4.4 for a particular case. Minimizing  $T$ , prevents the full exploitation of large propagation delays in UWA networks. To utilize the degree of freedom that is provided by varying the packet duration, we formulate a different objective which results in variable packet duration and eventually a significant gain in throughput. We know the throughput upper bound is  $\frac{N}{2}$  from [1], i.e.,

$$S \leq \frac{N}{2} \Rightarrow N - 2S \geq 0 \quad (4.15)$$

$$\Rightarrow \frac{NT - 2 \sum_{j=1}^N \left[ \sum_{(k,j) \in \mathcal{L}} \sum_{x=1}^{\Lambda_{kj}} \tau_{kj}^x \right]}{T} \geq 0. \quad (4.16)$$

The numerator in the resulting equation shown in (4.16) is split into three terms to show that it results in fractional idle time in a frame as shown in (4.17). Note that the minimization of fractional idle time results in maximal usage of the total time available in a frame and allows for multiple nodes to transmit simultaneously without causing collisions. The objective function to be

minimized as the fractional idle time in a frame is:

$$f_{\text{MILFP}} = \frac{1}{T} \left[ NT - \sum_{j=1}^N \left\{ \underbrace{\sum_{\forall(j,k) \in \mathcal{L}} \sum_{x=1}^{\Lambda_{jk}} \tau_{jk}^x}_{\text{sum of TX Packet duration from Node } j} + \underbrace{\sum_{\forall(k,j) \in \mathcal{L}} \sum_{x=1}^{\Lambda_{kj}} \tau_{kj}^x}_{\text{sum of RX Packet duration at Node } j} \right\} \right] \quad (4.17)$$

where  $N$  is the total number of nodes in the network. The first term of the numerator in (4.17) is the sum total time available including all the  $N$  nodes in the network per frame. The rest comprises of two terms which constitute the amount of time the network nodes are busy with transmissions and receptions. The frame length  $T$  in the denominator of (4.17) prevents the frame length from taking very small values resulting in trivial solutions. The objective function in (4.17) is a ratio of two linear functions. This is a non-linear objective function and together with the mixed-integer propagation delay constraints the problem is a Mixed-Integer Linear Fractional Problem (MILFP) [97]:

$$\begin{aligned} \min \quad & \frac{1}{T} \left[ NT - \sum_{j=1}^N \left\{ \sum_{(j,k) \in \mathcal{L}} \sum_{x=1}^{\Lambda_{jk}} \tau_{jk}^x + \sum_{(k,j) \in \mathcal{L}} \sum_{x=1}^{\Lambda_{kj}} \tau_{kj}^x \right\} \right] \\ \text{s.t.} \quad & t_{jk}^x - t_{li}^y + \tau_{jk}^x - Mp_{jk,li}^{xy} \leq -(D_{ji} - D_{li}) \\ & -t_{jk}^x + t_{li}^y + \tau_{li}^y + Mp_{jk,li}^{xy} \leq (D_{ji} - D_{li}) + M \\ & t_{jk}^x - t_{li}^y + \tau_{jk}^x + T - Mq_{jk,li}^{xy} \leq -(D_{ji} - D_{li}) \\ & -t_{jk}^x + t_{li}^y + \tau_{li}^y - T + Mq_{jk,li}^{xy} \leq (D_{ji} - D_{li}) + M \\ & t_{jk}^x - t_{li}^y + \tau_{jk}^x - T - Mr_{jk,li}^{xy} \leq -(D_{ji} - D_{li}) \\ & -t_{jk}^x + t_{li}^y + \tau_{li}^y + T + Mr_{jk,li}^{xy} \leq (D_{ji} - D_{li}) + M \\ & -t_{jk}^{x+1} + t_{jk}^x + \tau_{jk}^x \leq 0 \\ & t_{jk}^x + \tau_{jk}^x - 2T < -G. \end{aligned} \quad (4.18)$$

---

**Algorithm 2** Parametric Algorithm

---

**Initialization.**

Set  $n = 1, w_1 := 0, \epsilon = 10^{-4}$

**for**  $n = 1, 2, \dots$  **do**

Solve equivalent sub-problem

$F(w) = \min \left\{ \mathbf{c}^T \mathbf{x} + \mathbf{d} - w_n(\mathbf{e}^T \mathbf{x} + f) \right\}$  subject to constraints in (4.18)

**if**  $|F(w_n)| \geq \epsilon$  **then**

let  $w_{n+1} = \frac{\mathbf{c}^T \mathbf{x}_n^* + \mathbf{d}}{\mathbf{e}^T \mathbf{x}_n^* + f}$

**else**

Stop, output  $x_n^*$  as optimal solution and  $w^* = w_n$  as the optimal value

**end if**

**end for**

---

The domain of the variables in the above optimization problem are  $t_{jk}^x, T \in \mathbb{R}^+$  and  $p_{jk,li}^{xy}, q_{jk,li}^{xy}, r_{jk,li}^{xy} \in \{0, 1\} \forall j, k, l, i \in \{1, \dots, N\}, (j, k), (l, i) \in \mathcal{L}, x \in \{1, \dots, \Lambda_{jk}\}, y \in \{1, \dots, \Lambda_{li}\}$  and  $\tau_{lb} \leq \tau_{jk}^x \leq \infty$ , where  $\tau_{lb} \in \mathbb{R}^+$  is the lower bound set on the packet duration.

This formulation does not restrict the packet duration from being zero. This might lead to unfair distribution of time with most time being assigned to the nodes in the network such that the throughput is maximized. To ensure some link fairness, a lower bound  $\tau_{lb}$  on the packet duration for any transmission on link  $(j, k)$  can be set to a non-zero value. This ensures that node  $j$  transmits to node  $k$  atleast  $\Lambda_{jk} \tau_{lb}$  s per frame.

### 4.3 MILFP Solution

A MILFP has an objective function which is a ratio of two linear functions subject to mixed-integer linear constraints. The parametric algorithm based on Newton's method has been recently proposed as an efficient solution method to MILFP problems [97], [98]. For the sake of brevity, let us denote the objective

function in (4.17), which is a ratio of two linear functions, in a concise form as:

$$f_{\text{MILFP}} = \frac{1}{T} \left[ NT - \sum_{j=1}^N \left\{ \sum_{(j,k) \in \mathcal{L}} \sum_{x=1}^{\Lambda_{jk}} \tau_{jk}^x + \sum_{(k,j) \in \mathcal{L}} \sum_{x=1}^{\Lambda_{kj}} \tau_{kj}^x \right\} \right] = \frac{\mathbf{c}^T \mathbf{x} + \mathbf{d}}{\mathbf{e}^T \mathbf{x} + f} \quad (4.19)$$

where  $\mathbf{x}$  is the vector of variables and  $\mathbf{c}, \mathbf{e}$  are coefficients in the fractional linear objective.

The main idea of this algorithm is to transform the original MILFP problem into an equivalent parametric MILP problem  $F(w)$  as shown in Algorithm 2. This problem has the same constraints but a different objective function formulated as the numerator of the original objective function minus the denominator multiplied by a parameter,  $w$ . One unique feature of the function  $F(w)$  is that when  $F(w) = 0$ , the inner MILP problem has a unique optimal solution which is exactly the same as the global optimal solution to the original MILFP problem [98]. Based on this property of  $F(w)$ , solving the MILFP problem becomes equivalent to finding the root of the equation  $F(w) = 0$ . Therefore, numerical root-finding approaches such as Newton's method can be applied to solve this problem.

#### 4.4 Results

To demonstrate throughput gain, we first consider a realistic network geometry from an at-sea experiment and then show the average throughput gain computed over several random 3 node network deployments. In order to show the throughput gain that can be achieved by packet duration variability, we present a comparative study between the following methods:



Figure 4.2: UNET network node locations during the MISSION 2013 experiment (deployment #1). White markers are network nodes. The geometry considered for the case study is marked with the distances between the links.

- *Time-slotted fixed packet duration solution* – In this approach, we compute the time-slotted transmission schedule to be used with optimal time slot length and packet duration minimizing the guard times. We use the algorithm presented in [1] to compute the throughput optimal schedule for the considered network geometry.
- *Unslotted fixed packet duration solution* – For this case, we consider the MILP algorithm presented in [2] and compute throughput at the optimal value of packet duration which is an unslotted schedule with the least frame length.
- *Variable packet duration with no time slotting solution* – In this case, we solve the MILFP shown in (4.18), which minimizes the fractional idle time in a frame, using Algorithm 2 and calculate the throughput.

#### 4.4.1 Throughput gain - Sea-trial network geometry

The UNET network deployed (Fig. 4.2) during the MISSION 2013 experiment in Singapore waters consisted of a UNET-II modem [99] (marked P21 in Fig. 4.2) mounted below a barge and six UNET-PANDA nodes [100] deployed at various locations within  $2 \times 2$  km area around the barge. The modems labeled as P21, P28 and P29 in Fig. 4.2 are node 1, node 2 and node 3 respectively in the analysis. Given this network geometry, we schedule the links in  $\mathcal{L} = \{(1, 2), (2, 1), (2, 3), (3, 2), (1, 3), (3, 1)\}$ . The corresponding propagation delays are computed considering the speed of sound underwater  $c = 1540$  m/s. For the considered network geometry the distances among the nodes are marked as shown in Fig. 4.2.

#### Time-slotted fixed packet duration solution

In order to compute the schedules using algorithm presented in [1], we require the propagation delay between links in units of time slot length. Moreover, the algorithm in [1] only accepts integer propagation delays as explained in Chapter 3. The preliminary concepts and notation required to understand the solution are already presented in Section 2.3. If the network has a non-integer delay matrix, packets transmitted on the time slot boundaries may be received across time slot boundaries. For a non-integer delay matrix  $\mathbf{D}$ , the elements are rounded off to yield an integer delay matrix  $\mathbf{D}'$  and the largest round off errors in approximating  $D_{jk}$ 's to the nearest smaller and larger integer are denoted by  $\rho^+$  and  $\rho^-$  respectively and defined in (2.3) and (2.4).

Due to the approximations in the delay matrix, the guard intervals are

needed at the start and end of the time slots.  $\rho^+$  and  $\rho^-$  are the worst delay-approximations made. The packet transmission on the links with these delay-approximations will either yield in *early reception* of the packet or a *delayed reception* of the packet depending on whether propagation delay (in units of time slot length  $\tau$ ) of that link is approximated to a larger number or a smaller number respectively. It is obvious that  $\tau\rho^-$  is the worst amount of time that must be left before the transmission starts in order to prevent the early receptions and  $\tau\rho^+$  is the worst amount of time that must be left after the transmission ends in the time slot, in order to prevent the delayed reception. Hence, we denote these start and end guard times in the time slot by:

$$t_s = \tau\rho^- \tag{4.20}$$

$$t_e = \tau\rho^+. \tag{4.21}$$

Therefore, the maximum duration for which the time slot can be used for transmission is given by,

$$\tau - (t_s + t_e) = \tau(1 - \rho^- - \rho^+). \tag{4.22}$$

An exhaustive search by varying the time slot length from  $\tau_{\min}$  to  $\tau_{\max}$  in steps of size  $\Delta x$  will provide the optimal time slot length  $\tau$  and guard times  $t_s, t_e$ , that can be used.  $\Delta x$  is the smallest incremental duration in the packet length. In practical modems, the step size  $\Delta x$  depends on many factors like modulation and coding scheme employed at the physical layer. We set  $\Delta x = 1$  ms while

TABLE 4.1: TIME-SLOTTED FIXED PACKET DURATION SOLUTION

Parameter	Value
$\tau^*$	204 ms
$\tau^*(1 - \rho^- - \rho^+)$	184 ms
$t_s$	19.03 ms
$t_e$	0.97 ms
$S_\rho$	1.35

performing the exhaustive search. In practice  $\Delta x$  is usually larger than 1 ms (e.g., in the UNET modem [99]), and hence our throughput estimate here is intentionally optimistic.  $\tau_{\min}$  and  $\tau_{\max}$  are the minimum and maximum possible time slot lengths, the values of which are constrained by the minimum or maximum packet duration that can be set in the underwater acoustic modems. The throughput  $S_\rho$  is then computed using (2.10).  $S_\rho$  is the number of successful transmissions per time slot multiplied by the time slot efficiency. The time slot length corresponding to the maximum value of the objective function is chosen to be the optimal time slot length  $\tau^*$  and corresponding packet duration is set. Throughput sensitivity to the selection of time slot length is shown in Fig. 4.3 when schedules are computed using algorithm proposed in [1]. Time slot length is varied from 1 ms to 1000 ms and for those time slot lengths which result in a good approximation to integer delay matrix, the schedule is computed and the corresponding throughput is plotted in the case of [1]. The optimal time slot length and the corresponding throughput is marked in Fig. 4.3.

The time slot length  $\tau$  is varied from  $\tau_{\min} = 1$  ms to  $\tau_{\max} = 1000$  ms and  $\Delta x = 1$  ms is set. The optimal value of time slot length and other parameters are tabulated in Table 4.1. The delay matrix and the integer delay matrix

corresponding to time slot length  $\tau^*$  are:

$$\mathbf{D} = \frac{\mathbf{L}}{c\tau^*} = \begin{bmatrix} 0 & 1.9067 & 2.9666 \\ 1.9067 & 0 & 3.0048 \\ 2.9666 & 3.0048 & 0 \end{bmatrix}, \quad \mathbf{D}' = \begin{bmatrix} 0 & 2 & 3 \\ 2 & 0 & 3 \\ 3 & 3 & 0 \end{bmatrix}.$$

With this delay matrix, the optimal schedule is computed using the algorithm presented in [1]. The frame length computed is  $T = 8$  slots with each slot duration  $\tau^* = 204$  ms. The throughput is calculated to be  $S_\rho = 1.35$ . The schedule is visualized in Fig. 4.5, using the optimal values computed for the network setting. Since we know the time slot length, guard times and frame length of the schedule, we can plot the transmitted packets, received packets and interfered packets accurately in time at each node. We leave  $t_s$  amount of time before the start of the transmission in a transmitting slot and  $t_e$  amount of time at the end of the transmission. In Fig. 4.5, it is clear that all the receptions are interference-free as expected and all interfering packets are aligned with the transmitting slots.

### Unslotted fixed packet duration solution

In [2], the scheduling problem is formulated as a MILP and the objective function considered for minimization is frame length  $T$ . The packet duration is fixed. In order to find the optimal packet duration resulting in the maximum throughput we vary the packet duration similar to the time slot length in Section 4.4.1 in steps of 1 ms. For each value of fixed packet duration set, we compute the schedule and the corresponding throughput. The throughput computed

TABLE 4.2: UNSLOTTED FIXED PACKET DURATION SOLUTION

Link	Transmission Start Time (s)	Packet Duration (s)
(1, 2)	3.2676	0.539
(2, 1)	2.0495	0.539
(2, 3)	0.1422	0.539
(3, 2)	0.0683	0.539
(1, 3)	1.3604	0.539
(3, 1)	3.7405	0.539

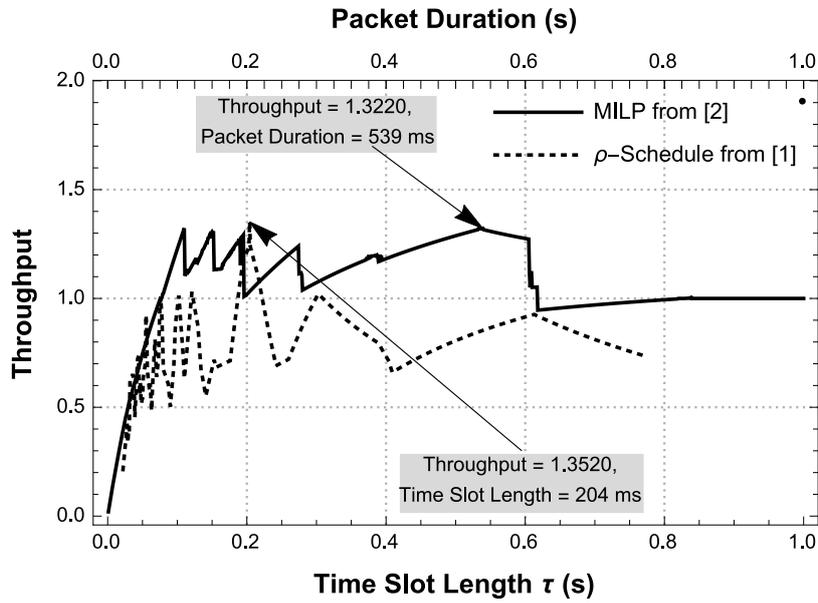


Figure 4.3: Throughput sensitivity to time slot length while using the algorithm in [1] and to packet duration while using the algorithm in [2].

is plotted against the packet duration values and shown in Fig. 4.3. The optimal packet duration and the corresponding throughput is marked in Fig. 4.3. The transmission start times in the schedule at the optimal packet duration is tabulated in Table 4.2. The frame length is  $T = 2.4462$  s. Throughput  $S$  is computed from (4.13),  $S = 1.3220$ . The optimal schedule computed is visualized in Fig. 4.6. To visualize the schedule we plot the transmission and reception events in time based on the transmission times, propagation delay information and the packet duration computed. Also, note that all the receptions in the

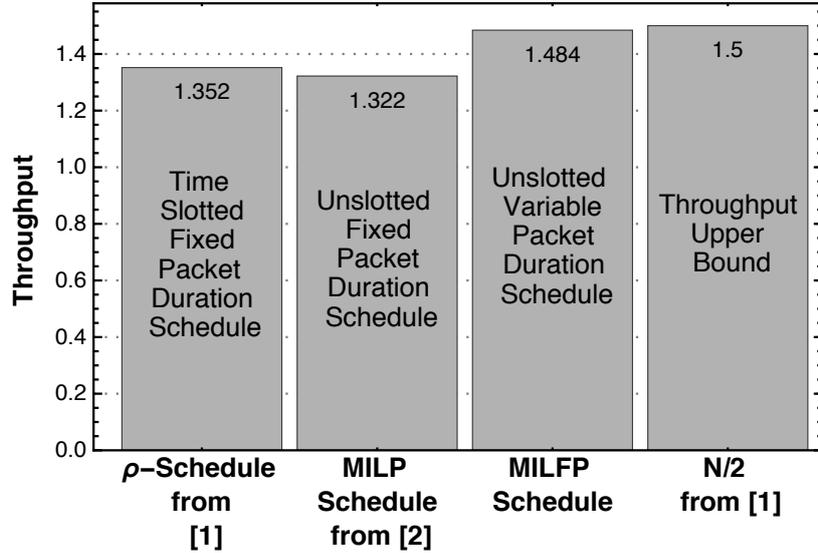


Figure 4.4: Throughput comparison when using proposed MILFP technique resulting in unslotted variable packet duration schedules for the considered sea-trial network geometry.

schedule shown in Fig. 4.6 are interference-free and most of the interference aligns with the transmission times. The lower bound on the packet duration is set to different values and the solution still results in all transmissions with equal value, same as the lower bound set.

Note that even if the packet duration is considered a variable in the MILP in [2], the packet duration values in the resulting solution are equal to the lower bound set on the packet duration. The reason for all the transmission duration to be same in this case is due to the choice of the objective function which is the frame length  $T$ . If the frame length is minimized, the packet duration of all transmissions contributing to the frame length have to be the least possible values that they can have. Therefore, frame length is not a good choice if the potential of exploiting the large propagation delays need to be studied, since this

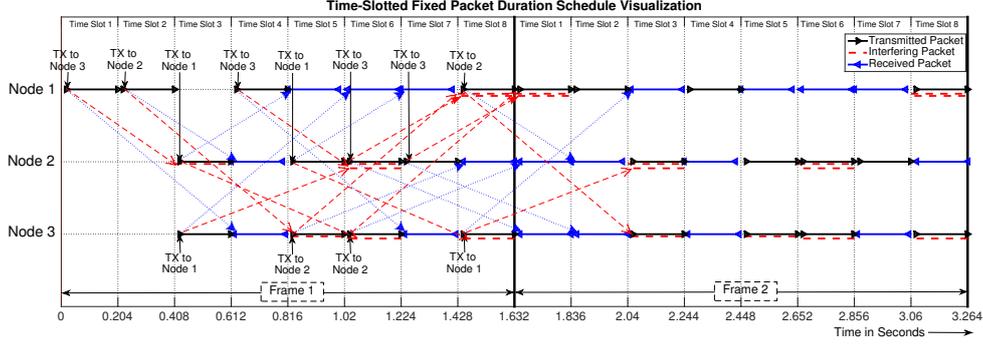


Figure 4.5: Schedule visualization with fixed time slot length and packet duration. The guard times set in the time slot take care of the early receptions and delayed receptions problem due to the approximations made in the delay matrix. Throughput achieved is 1.35.

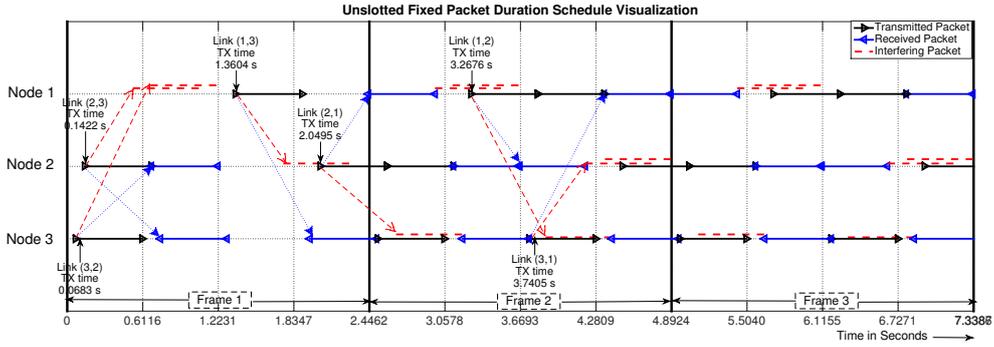


Figure 4.6: Schedule visualization for unslotted fixed packet duration schedule. Note that all the packet transmissions are of equal size - 539 ms. Throughput computed is 1.32.

formulation will not allow the variability in the packet duration.

### Variable packet duration with no time slotting solution

Next, we consider the solution resulting from solving (4.18) using Algorithm 2.

- *Uniform packet traffic demand:* The lower bound on packet duration  $\tau_{jk}^x$  is set to 0 and the packet traffic demand considered is 1 packet/frame on each link. For the considered setting, the transmission times and corresponding packet duration are tabulated in Table 4.3. The frame length computed

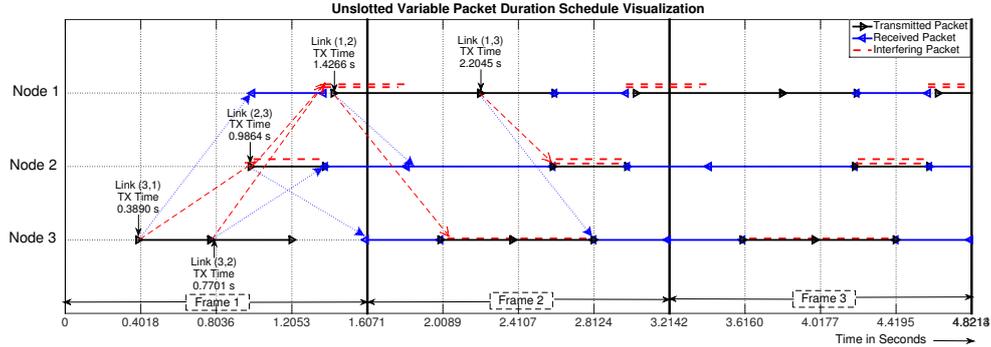


Figure 4.7: Schedule visualization with variable packet duration. Note that there are no time slots and the transmission times and packet lengths are such that the total idle time is minimized. Throughput achieved is 1.484, which is significantly closer to the upper bound on the throughput.

is  $T = 1.6071$  s. Throughput,  $S$  is computed from (4.13),  $S = 1.484$ .

Note that, the throughput computed in this case is significantly closer to the upper bound  $\frac{N}{2}$  (which is 1.5) as compared to the throughput computed in the previous section with fixed packet duration and time slot length as shown in Fig. 4.4. The reason for an increase in the throughput from the previous case is the variability in the packet duration. From Fig. 4.7, it is clear that by varying the packet duration the transmissions and receptions on each node can be scheduled in such a way that the total idle time is minimized while exploiting the large propagation delays. This results in the throughput gain. Note that all the receptions in Fig. 4.7 are interference-free and the interfering packets are aligned with the duration in which the nodes are busy transmitting. However, the packet traffic demand was set to be one packet for each link for this example. The schedule found is optimal given the network geometry and the packet traffic demand. Do note that one of the packet duration on the link (2, 1) is 0. This essentially

TABLE 4.3: VARIABLE PACKET DURATION SOLUTION

Link	Transmission Start Time (s)	Packet Duration (s)
(1, 2)	1.4266	0.7779
(2, 1)	0.9864	0
(2, 3)	0.9864	0.3968
(3, 2)	0.7701	0.4325
(1, 3)	2.2045	0.3968
(3, 1)	0.3890	0.3812

TABLE 4.4: VARIABLE PACKET DURATION SOLUTION WITH PACKET HEADERS

Link	Transmission Start Time (s)	Payload Duration (s)
(1, 2)	0.0278	0.7579
(2, 1)	1.1947	0.0157
(2, 3)	1.2304	0.3768
(3, 2)	1.0142	0.3768
(1, 3)	0.8057	0.4125
(3, 1)	0.6330	0.3255

means that the transmission on this link does not take place. We did not constraint the problem in (4.18) with the requirement for any minimum duration for which each link needs to be served. This can be easily added by setting a positive lower bound on the packet duration.

- *Non-negligible packet headers:* Next, let us consider including the packet headers in the problem as explained in Section 4.2.5. Note that inclusion of packet headers is needed if the packet duration values lesser than the packet header lengths is to be prevented in the resulting solution. The provision to include this in the problem allows us to find schedules such that the packet duration in the solution is at least greater than the packet header lengths. The propagation delay constraints (4.3) & (4.4) after including

packet header  $\tau_p$  are as follows:

$$t_{jk}^x + \beta T + (\tau_p + \tau_{jk}^x) + D_{ji} \leq t_{li}^y + D_{li} \quad (4.23)$$

(or)

$$t_{jk}^x + \beta T + D_{ji} \geq t_{li}^y + (\tau_p + \tau_{li}^y) + D_{li} \quad (4.24)$$

$\forall \{(j, k), (l, i) \in \mathcal{L}\}$ . The constraints are transformed and the corresponding MILFP is setup. The objective function need not be changed since the total time contributed by the packet headers of each transmission and reception is considered as the idle time and is a constant for known packet header duration.

We set the packet header duration to be 20 ms and leave all other settings same as in the previous section. The optimal solution with transmission times and the packet duration is computed and tabulated in Table 4.4. The frame length in the solution is expected to be no lesser than its value computed without considering the packet headers. In this case, the frame length computed remains same,  $T = 1.6071$  s, however, with a different schedule as can be observed from the value of transmission times and the corresponding packet duration. The throughput is calculated,  $S = 1.4094$ . Note that this throughput should not be compared against the upper bound 1.5, since with the inclusion of packet headers, this upper bound can never be achieved. However, if we consider the channel utilization, the total packet duration can be considered and can be compared to 1.5. The total

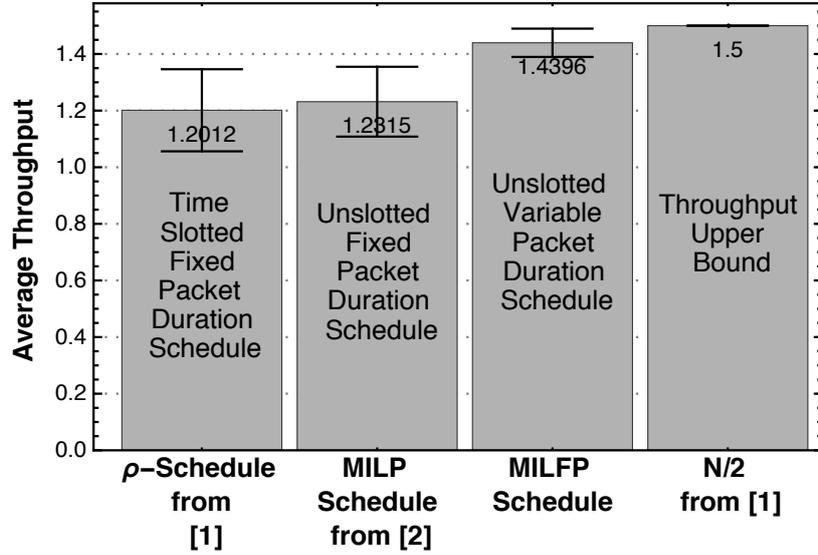


Figure 4.8: Average throughput is computed over 100 network geometries with 3 nodes and 6 links to compare against the known upper bound on the throughput.

channel utilization is 1.484 which is much closer to the upper bound. Also note that for this case the channel utilization remains the same to the case without the packet header consideration, however with a different schedule (see Table 4.4). The packet duration for the transmission on link (2,1) in the case without the packet headers was 0 (see Table 4.3), while after considering the packet headers, on the same link (2,1), the payload duration is 15.7 ms.

$S_\rho$ , with the inclusion of packet headers, is computed by modifying (2.10) to consider only payload duration:

$$S_\rho = \frac{Z}{T} \left( 1 - \rho^- - \rho^+ - \frac{t_p}{\tau} \right). \quad (4.25)$$

The optimal values of time slot length and other parameters are computed similarly as explained in Section 4.4.1. The throughput  $S_\rho = 1.205$  is computed with  $\tau_p = 20$  ms as compared to  $S = 1.4094$  with variability in packet duration. This shows that the solution provided by MILFP (4.18), better utilizes the large propagation delay in UWA networks.

To summarize the results, the computed throughput for the considered sea-trial network geometry using the algorithms from [1], [2] and MILFP (4.18) are shown in Fig. 4.4. Note that allowing variability in the packet duration resulted in a throughput much closer to the upper bound 1.5.

#### 4.4.2 Throughput gain - Randomly deployed network geometries

To generalize the results obtained for the 3 node sea-trial network geometry, we consider an equilateral triangle network for which the optimal schedule and throughput is known from [1] and randomly perturb the locations of the nodes in the network within a circle of radius  $r$ . For an equilateral triangle network, the optimal throughput is 1.5 and can be achieved when the optimal schedule is adopted [1]. However, the random perturbations in the location of nodes effect the throughput. The propagation delay on each of the link is set to 1 s. The perturbations in the node locations causing a change in the propagation delay of at most 0.2 s is considered, i.e., the propagation delay on each link may lie between 0.8 s to 1.2 s and is uniformly chosen in each instance. The algorithms from [1], [2] and MILFP (4.18) are used to compute the schedules and the corresponding throughput. The average throughput is computed over 100 such random instances. The comparison is shown in Fig. 4.8. Note that

the results from [1] and [2] almost remain the same, however, the significant gain in the throughput is observed when the variability in the packet duration is allowed. The error bars using the standard deviation of the throughput values is also shown. Note that the average throughput computed using MILFP algorithm results in greater throughput with a smaller deviation and it is evident that the variability in the packet duration causes the network throughput to be much closer to the upper bound.

#### 4.5 Network Geometries with Arbitrary Traffic Demands

In this section, we present optimal schedules for some network geometries, given arbitrary packet traffic demands with an objective to study the effect of packet traffic demands and to provide insights by comparing the similarities and differences with schedules computed in [1]. We elucidate these objectives here:

- *Effect of packet traffic demand:* The throughput optimal schedules are already known for the network geometries that are considered here, from [1]. We study the effect of including the traffic demands on these throughput optimal schedules. This setup is more practical, since, in reality, the traffic demands are derived from user requirements. The results in [1] only select those links for transmission which are throughput-maximizing. However, the problem formulation in (4.18), allows us to select the links and the corresponding packet traffic demands. The lower bound on the packet duration can be set to a non-zero value to ensure minimum link fairness. This gives us control on the minimum fairness level to be maintained for all the links considered for scheduling.

We demonstrate these features of the algorithm in this section.

- *Schedule matrix representation of MILFP solution:* Since all the network geometries considered in this section for illustration have integer propagation delays among the links, the schedules found by solving (4.18) results in equal packet duration. Therefore, the solution can be represented in the form of a time-slotted schedule by simply setting the time slot length equal to the packet duration. This will be helpful in comparing the schedules computed with the time-slotted schedules in [1], which can be represented in the matrix form and are elucidated in Section 2.3. This provides insights on the similarity in the results obtained from [1] and the proposed MILFP algorithm. Note that solving the MILFP presented in (4.18) can be represented in the form of schedule matrix only because the solution for the considered network geometries in this section result in equal packet duration and thus provides an opportunity to compare the solutions and appreciate the similarity and differences among them.

#### 4.5.1 Illustrative network geometries

For each network geometry, we consider two different sets of packet traffic demands denoted by  $\Lambda^1$  and  $\Lambda^2$ . Each set  $\Lambda^1$  and  $\Lambda^2$  contains the packet traffic demand for each link  $l \in \mathcal{L}$ . For example,  $\Lambda_l^1$  packets/frame is the packet traffic demand on link  $l$  from the first set and  $\Lambda_l^2$  packets/frame is the packet traffic demand on link  $l$  from the second set. Table 4.5 lists the arbitrarily set packet traffic demands for the networks considered. For example  $\Lambda_{(1,2)}^2 = 3$  for the equilateral triangle network in Table 4.5 implies that, there must be three packet

4.5. NETWORK GEOMETRIES WITH ARBITRARY TRAFFIC DEMANDS

TABLE 4.5: ARBITRARY PACKET TRAFFIC DEMANDS FOR ILLUSTRATIVE NETWORK GEOMETRIES CONSIDERED

Link	Equilateral Triangle			Isosceles Triangle			Linear Network		
	Propagation Delay (s)	Packet Traffic Demand		Propagation Delay (s)	Packet Traffic Demand		Propagation Delay (s)	Packet Traffic Demand	
		$\Lambda_i^1$ pkts/frame	$\Lambda_i^2$ pkts/frame		$\Lambda_i^1$ pkts/frame	$\Lambda_i^2$ pkts/frame		$\Lambda_i^1$ pkts/frame	$\Lambda_i^2$ pkts/frame
(1,2)	1	1	3	1	1	1	1	1	2
(2,1)	1	1	1	1	1	2	1	1	1
(2,3)	1	1	2	2	1	1	1	1	2
(3,2)	1	1	2	2	1	2	1	1	1
(1,3)	1	1	1	2	1	1	2	1	2
(3,1)	1	1	3	2	1	2	2	1	1

TABLE 4.6: OPTIMAL SCHEDULES FOR EQUILATERAL TRIANGLE NETWORK GEOMETRY

Link	$\Lambda_i^1$ packets/frame		$\Lambda_i^2$ packets/frame	
	Transmission Time (s)	Packet Duration (s)	Transmission Time (s)	Packet Duration (s)
(1,2)	$t_{12}^1 = 0$	$\tau_{12}^1 = 1$	$t_{12}^1 = 3, t_{12}^2 = 6, t_{12}^3 = 7$	$\tau_{12}^1 = 1, \tau_{12}^2 = 1, \tau_{12}^3 = 1$
(2,1)	$t_{21}^1 = 0$	$\tau_{21}^1 = 1$	$t_{21}^1 = 3$	$\tau_{21}^1 = 1$
(2,3)	$t_{23}^1 = 3$	$\tau_{23}^1 = 1$	$t_{23}^1 = 1, t_{23}^2 = 5$	$\tau_{23}^1 = 1, \tau_{23}^2 = 1$
(3,2)	$t_{32}^1 = 1$	$\tau_{32}^1 = 1$	$t_{32}^1 = 1, t_{32}^2 = 5$	$\tau_{32}^1 = 1, \tau_{32}^2 = 1$
(1,3)	$t_{13}^1 = 2$	$\tau_{13}^1 = 1$	$t_{13}^1 = 2$	$\tau_{13}^1 = 1$
(3,1)	$t_{31}^1 = 2$	$\tau_{31}^1 = 1$	$t_{31}^1 = 0, t_{31}^2 = 4, t_{31}^3 = 8$	$\tau_{31}^1 = 1, \tau_{31}^2 = 1, \tau_{31}^3 = 1$
<p><b>Equivalent Schedule Matrix</b></p> $\mathbf{W}^{(4)} = \begin{bmatrix} 2 & -2 & 3 & -3 \\ 1 & -1 & -3 & 3 \\ -2 & 2 & 1 & -1 \end{bmatrix}$ <p><b>Throughput</b></p> $\Rightarrow S = \frac{6}{4} = 1.5$			<p><b>Equivalent Schedule Matrix</b></p> $\mathbf{W}^{(9)} = \begin{bmatrix} -3 & -3 & 3 & 2 & -2 & -3 & 2 & 2 & 0 \\ 0 & 3 & -3 & 1 & -1 & 3 & -3 & -1 & -1 \\ 1 & 2 & -2 & -1 & 1 & 2 & -2 & 0 & 1 \end{bmatrix}$ <p><b>Throughput</b></p> $\Rightarrow S = \frac{12}{9} = 1.33$	

transmissions on link (1,2) per frame.

CHAPTER 4. UNSLOTTED SCHEDULES WITH VARIABLE PACKET LENGTHS

TABLE 4.7: OPTIMAL SCHEDULES FOR ISOSCELES TRIANGLE NETWORK GEOMETRY

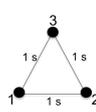
Link	$\Lambda_l^1$ packets/frame		$\Lambda_l^2$ packets/frame	
	Transmission Time (s)	Packet Duration (s)	Transmission Time (s)	Packet Duration (s)
(1, 2)	$t_{12}^1 = 0$	$\tau_{12}^1 = 1$	$t_{12}^1 = 0$	$\tau_{12}^1 = 1$
(2, 1)	$t_{21}^1 = 2$	$\tau_{21}^1 = 1$	$t_{21}^1 = 0, t_{21}^2 = 2$	$\tau_{21}^1 = 1, \tau_{21}^2 = 1$
(2, 3)	$t_{23}^1 = 3$	$\tau_{23}^1 = 1$	$t_{23}^1 = 3$	$\tau_{23}^1 = 1$
(3, 2)	$t_{32}^1 = 2$	$\tau_{32}^1 = 1$	$t_{32}^1 = 2, t_{32}^2 = 4$	$\tau_{32}^1 = 1, \tau_{32}^2 = 1$
(1, 3)	$t_{13}^1 = 1$	$\tau_{13}^1 = 1$	$t_{13}^1 = 6$	$\tau_{13}^1 = 1$
(3, 1)	$t_{31}^1 = 0$	$\tau_{31}^1 = 1$	$t_{31}^1 = 0, t_{31}^2 = 3$	$\tau_{31}^1 = 1, \tau_{31}^2 = 1$
<p>Equivalent Schedule Matrix</p> $\mathbf{W}^{(4)} = \begin{bmatrix} 2 & 3 & -3 & -2 \\ -3 & -1 & 1 & 3 \\ 1 & -2 & 2 & -1 \end{bmatrix}$ <p>Throughput</p> $\Rightarrow S = \frac{6}{4} = 1.5$			<p>Equivalent Schedule Matrix</p> $\mathbf{W}^{(7)} = \begin{bmatrix} 2 & -2 & -3 & -2 & 0 & -3 & 3 \\ 1 & -1 & 1 & 3 & -3 & 0 & -3 \\ 1 & -1 & 2 & 1 & 2 & -2 & 0 \end{bmatrix}$ <p>Throughput</p> $\Rightarrow S = \frac{9}{7} = 1.2857$	

TABLE 4.8: OPTIMAL SCHEDULES FOR LINEAR NETWORK GEOMETRY

Link	$\Lambda_l^1$ packets/frame		$\Lambda_l^2$ packets/frame	
	Transmission Time (s)	Packet Duration (s)	Transmission Time (s)	Packet Duration (s)
(1, 2)	$t_{12}^1 = 1$	$\tau_{12}^1 = 1$	$t_{12}^1 = 0, t_{12}^2 = 6$	$\tau_{12}^1 = 1, \tau_{12}^2 = 1$
(2, 1)	$t_{21}^1 = 1$	$\tau_{21}^1 = 0$	$t_{21}^1 = 0$	$\tau_{21}^1 = 1$
(2, 3)	$t_{23}^1 = 0$	$\tau_{23}^1 = 1$	$t_{23}^1 = 3, t_{23}^2 = 6$	$\tau_{23}^1 = 1, \tau_{23}^2 = 1$
(3, 2)	$t_{32}^1 = 2$	$\tau_{32}^1 = 0$	$t_{32}^1 = 1$	$\tau_{32}^1 = 1$
(1, 3)	$t_{13}^1 = 0$	$\tau_{13}^1 = 1$	$t_{13}^1 = 3, t_{13}^2 = 4$	$\tau_{13}^1 = 1, \tau_{13}^2 = 1$
(3, 1)	$t_{31}^1 = 0$	$\tau_{31}^1 = 1$	$t_{31}^1 = 3$	$\tau_{31}^1 = 1$
<p>Equivalent Schedule Matrix</p> $\mathbf{W}^{(3)} = \begin{bmatrix} 3 & 2 & -3 \\ 3 & 0 & -1 \\ 1 & -2 & -1 \end{bmatrix}$ <p>Throughput</p> $\Rightarrow S = \frac{4}{3} = 1.33$			<p>Equivalent Schedule Matrix</p> $\mathbf{W}^{(8)} = \begin{bmatrix} 2 & -2 & 0 & 3 & 3 & -3 & 2 & 0 \\ 1 & -1 & -3 & 3 & 0 & 0 & 3 & -1 \\ 0 & 2 & 0 & 1 & -2 & -1 & -1 & -2 \end{bmatrix}$ <p>Throughput</p> $\Rightarrow S = \frac{9}{8} = 1.12$	

### Equilateral triangle

Consider an equilateral triangle shaped network geometry with the propagation delay between the links set as 1 s. We consider two different sets of packet traffic demands as shown in Table 4.5. The delay matrix<sup>2</sup> for a three node equilateral triangle network geometry is:

$$\mathbf{D} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$


and the optimal schedules computed by solving (4.18) for the packet traffic demands  $\Lambda_i^1$  and  $\Lambda_i^2$  are represented in Table 4.6. Moreover, the solution is also represented in the form of the schedule matrix  $\mathbf{W}$  to compare it with the solution in [1].

1.  $\Lambda_i^1$  packets/frame: The six links considered must be scheduled once in the frame. The solution found is listed in the Table 4.6, the frame length computed is  $T = 4$ . The transmission times and packet duration listed in Table 4.6 is represented in the form of the schedule matrix denoted as  $\mathbf{W}^{(4)}$ . Note that, there are 6 positive entries per frame of length  $T = 4$  s indicating 6 successful transmissions per frame, and hence the throughput is computed as  $S = \frac{6}{4} = 1.5$ . The throughput optimal schedule computed in [1] is the same schedule as shown in Table 4.6 for the equilateral

---

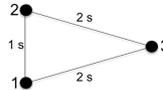
<sup>2</sup>Note, in this section, the elements of the delay matrix are in units of seconds and not in units of time slot length as considered in Section 4.4.1. We present the propagation delays in the form of the delay matrix only for maintaining uniformity with the representation in [1].

triangle where only the throughput-maximizing links are chosen. This shows that selection of  $\Lambda_l^1$  packet traffic demand is throughput optimal for the considered equilateral triangle network geometry.

2.  $\Lambda_l^2$  packets/frame: For this case, the six links are scheduled as many times as represented in the column  $\Lambda_l^2$  packets/frame of Table 4.5. The transmission times and packet duration is shown in Table 4.6 along with the equivalent schedule matrix  $\mathbf{W}^{(9)}$ . The frame length computed is  $T = 9$ . The formulation allows us to compute such schedules with arbitrary traffic demands which is not possible using the algorithm from [1]. Note that, the schedule computed has 3 idle slots and the throughput computed is 1.33, which is although greater than 1, is lesser than the throughput upper bound.

### Isosceles triangle

Next, we consider an isosceles triangle shaped network geometry with the delay matrix as shown below:

$$\mathbf{D} = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix}$$


We consider again two different sets of packet traffic demands as shown in Table 4.5.

1.  $\Lambda_l^1$  packets/frame: Again the six links considered must be scheduled once in the frame for this case. The frame length computed is  $T = 4$  and

throughput  $S$  is 1.5. The equivalent schedule matrix along with the MILFP solution is shown in Table 4.7. The schedule  $\mathbf{W}^{(4)}$  computed for this case is significantly different from the optimal schedule found for the same network geometry in [1]. Although both the schedules are throughput optimal, the difference is the frame length.

2.  $\Lambda_i^2$  packets/frame: Next, we consider each of the six links to be scheduled as per the packet traffic demand listed in  $\Lambda_i^2$  column of Table 4.5. The lower bound ( $\tau_{lb}$ ) on the packet duration is set to 1 s for this case to ensure that each link transmits at least 1 s. We expect to find schedules in which the packet traffic demand is satisfied along with the minimum packet duration requirement. The frame length is computed to be  $T = 7$ , and the optimal solution is represented in Table 4.7. Note that, all the six links considered are active for 1 s and the packet traffic demand is also satisfied, i.e., the links  $(2, 1)$ ,  $(3, 2)$  and  $(3, 1)$  each transmit twice in the frame along with the rest of the links, which transmitted one packet per frame as required.

### Linear network

Now we consider a 3 Node linear network with the delay matrix:

$$\mathbf{D} = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix}$$


The different packet traffic demands are shown in Table 4.5. The corresponding schedule is computed again by solving (4.18).

1.  $\Lambda_l^1$  packets/frame: For this case, we compute the schedules with both  $\tau_{lb} = 0$  and 1 s. In Table 4.8, we present the schedule with lower bound  $\tau_{lb}$  set to 0. When  $\tau_{lb}$  is set to 1 s, the optimal schedule matrix computed is:

$$\mathbf{W}^{(6)} = \begin{bmatrix} 2 & -2 & 3 & 3 & -3 & -3 \\ 1 & -1 & -3 & 0 & 0 & 3 \\ -2 & 2 & 1 & 1 & -1 & -1 \end{bmatrix} \Rightarrow S = \frac{8}{6} = 1.33. \quad (4.26)$$

The schedule computed is significantly different from the result in [1]. The schedule from [1] for the same linear network is same as shown in Table 4.8. Note that, for the schedule from [1], the links (2, 1) and (3, 2) are not scheduled. The reason is, the algorithm in [1], only selects the throughput-maximizing links. However, we solve the MILFP (4.18), by setting  $\tau_{lb} = 1$  s and  $\Lambda_l^1$  packets/frame on each link, to make sure that all the links are scheduled atleast once with packet duration of at least 1 s. Therefore, we see that links (2, 1) and (3, 2) are scheduled with a packet duration of atleast 1 s in  $\mathbf{W}^{(6)}$  as shown in (4.26).

2.  $\Lambda_l^2$  packets/frame: Similarly, we also present the MILFP solution and the equivalent schedule matrix with  $\tau_{lb} = 1$  s for this case as shown in Table 4.8. The frame length computed is  $T = 8$ , with throughput  $S = 1.12$ .

These examples demonstrate that the control provided by this formulation is necessary to compute useful schedules for many application scenarios with different traffic demands.

## 4.6 Scalability & Complexity Analysis

In Section 4.4, we considered an UWA network with single collision domain and demonstrated that unslotted schedules with variable packet duration result in throughput closer to the upper bound and outperform the state-of-the-art centralized algorithms presented in [1] and [2]. The proposed MILFP is a centralized algorithm and schedules are computed offline. The maximum network size (in terms of the number of nodes and links) for which MILFP can be solved in a reasonable amount of time is of interest. In order to study the scalability of the algorithm with the number of nodes and links in the network, we compute schedules for much larger multi-hop networks. The proposed algorithm is general enough to be used for finding schedules for multi-hop UWA networks with multiple partially overlapping collision domains. We consider multi-hop multi-line grid networks as shown in Fig. 4.9 (also considered in [3]) and compute the schedules using (4.18) as the number of nodes in the network increases. Multi-hop multi-line grid topology consists of parallel lines with regularly placed nodes. Messages originate from the first node on each line and are destined to the final node on the same line. Intermediate nodes act as relay nodes which receive the incoming packets, decode them and retransmit them to the next hop until they reach the final destination node. The spacing between neighboring nodes on the same line corresponds to a propagation delay of 1 s and the distance separating every two adjacent lines correspond to the propagation delay of 2 s (see [3] for details). The ratio of interference range to the communication range  $\alpha$  is set to 2 and hence a transmission on a link with propagation delay 1 s among

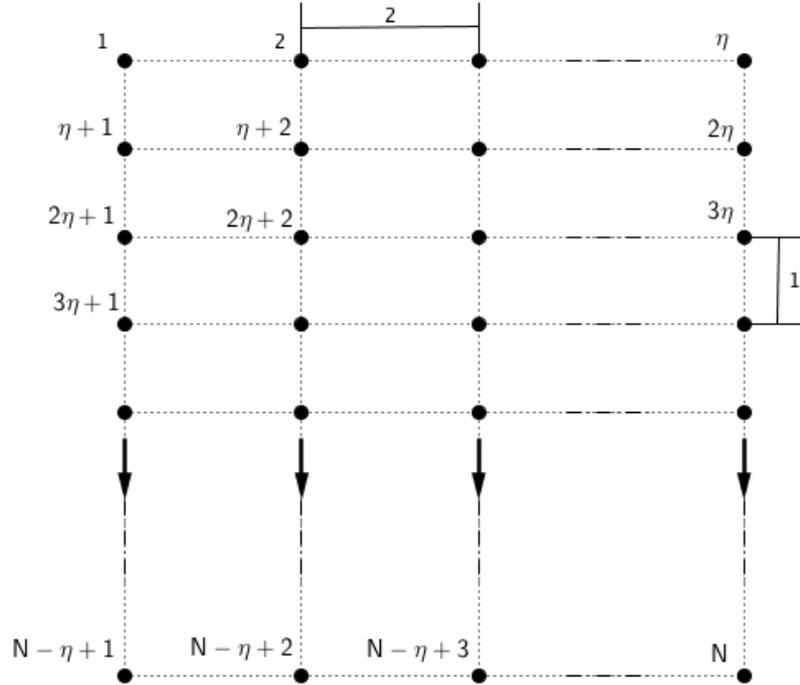


Figure 4.9: Regular  $N$ -node multi-line grid network with uniformly spaced nodes.

them interferes with the node on the adjacent line to which the propagation delay is 2 s. For a multi-hop network, multiple partially overlapping collision domains exist and the problem formulation presented in Section 4.2 allows us to enumerate the propagation delay constraints for such case.

The classical approach for solving MILP with binary variables used in Algorithm 2, is the tree search by a Branch & Bound algorithm with linear programming relaxation, which generally has an exponential complexity in the worst case. An indicator that can be quantified for indicating the complexity of the problem formulated is the number of binary variables  $b$  required. We can compute  $b$  in terms of number of nodes  $N$ , number of links to be scheduled  $|\mathcal{L}|$  and the packet traffic demand  $\Lambda_{jk}$  on each link  $(j, k) \in \mathcal{L}$ . The number of binary variables depends on the number of link pairs  $(j, k), (l, i) \in \mathcal{L}$  such that

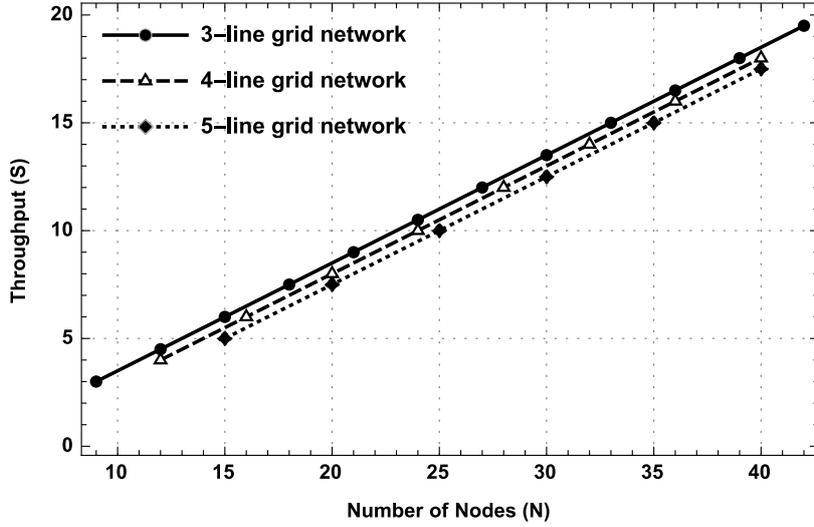


Figure 4.10: Scalability with number of nodes and links in a multi-hop multi-line grid network using MILFP.

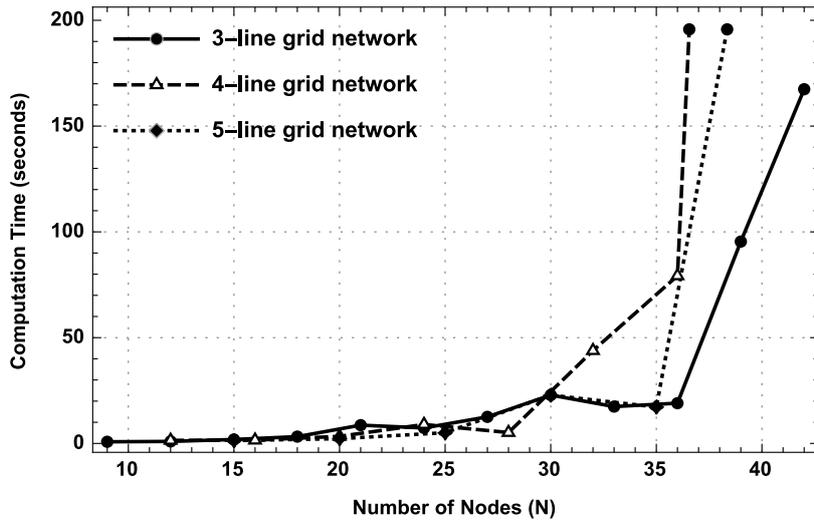


Figure 4.11: Computation time with number of nodes and links in a multi-hop multi-line grid network using MILFP.

the condition  $D_{ji} \leq \alpha D_{jk}$  is satisfied. For a single collision domain network, any transmission on a particular link  $(j, k)$  causes interference to all other nodes.

Therefore, the case of single collision domain presents the worst case for which

the number of binary variables is computed as follows:

$$b = (2\beta + 1) \left( \sum_{\forall (j,k) \in \mathcal{L}} \left( \sum_{\substack{\forall (l,i) \neq (j,k) \\ \in \mathcal{L}}} \Lambda_{jk} \Lambda_{li} \right) \right) \quad (4.27)$$

where  $\beta$  takes the maximum value for which the propagation delay constraints are enumerated in the MILFP (4.18). It is worth noting that the number of binary variables  $b$  scale on the order of the square of the number of links in  $\mathcal{L}$ . To see this, consider the case of a single collision domain network with packet traffic demand  $\Lambda_{jk} = 1$  packet/frame  $\forall (j, k) \in \mathcal{L}$ . For this case, with  $\beta = 1$  and using (4.27), we compute the number of binary variables to be

$$b = 3|\mathcal{L}|(|\mathcal{L}|-1). \quad (4.28)$$

We vary the number of nodes from 9 to 42 for the grid networks considered with 3 lines, 4 lines and 5 lines. Note that for an  $N$  node multi-hop multi-line grid network with  $\eta$  parallel lines there are  $N - \eta$  links to be scheduled. The MILFP (4.18) is solved for a minimum of 6 links to a maximum of 39 links. The optimal solution computed always corresponds to the maximum achievable throughput (see Fig. 4.10) for the network and can be verified from the results in [3]. For the case with  $N = 42$  nodes and 39 links, the number of binary variables  $b$  is computed to be 549, the number of constraints listed were 1176 and the total number of variables in the problem were 629. We use MOSEK optimizer with MATLAB on an iMac with 2.5 GHz Intel Core i5 quad-core processor to solve the MILP in Algorithm 2. The computation time in seconds for each

#### **4.7. REGION OF OPERATION - EXPLOITING LARGE PROPAGATION DELAYS**

---

case is plotted in Fig. 4.11 and can be seen that the optimal schedules are computed in reasonable time for all these networks. Note that these times only give a rough idea of time taken to solve the MILFP on a particular machine. For 42 nodes with 39 links the MILFP converges to the optimal solution in approximately 167 seconds. Also, note that in practice UWA networks are not as large as considered in this analysis and hence the proposed centralized algorithm provides a good alternative for providing better benchmarks in computing throughput-maximizing schedules.

#### **4.7 Region of Operation - Exploiting Large Propagation Delays**

In the previous sections, we have demonstrated that allowing variability in the packet duration significant throughput gains can be achieved and presented that proposed MILFP outperforms other state-of-the-art centralized algorithms. However, under what conditions can we exploit large propagation delay to achieve throughput gains? In this section, we show that only when the packet duration used for transmissions is comparable to the link propagation delay in the network, achievable throughput is greater than what can be achieved in the terrestrial wireless networks with negligible propagation delay. We study the region in which the network must be operated in, to take advantage of the large propagation delay.

##### **Proposition**

Assuming the frame length of a schedule to be greater than the girth of a single collision domain network, i.e.,  $T > G$ , the upper bound on the throughput of any arbitrary network geometry is found by solving (4.18) with the lower bound

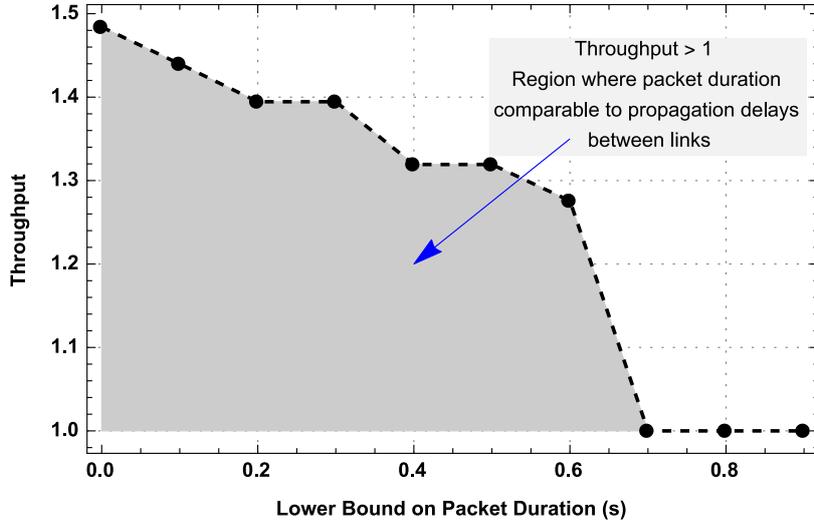


Figure 4.12: Sea-trial network geometry.

$\tau_b$  on the packet duration set to 0.

### Proof

Let the feasible set for the problem (4.18) be denoted by  $\mathcal{F}$  when  $\tau_b = 0$ , and denote the feasible set by  $\mathcal{F}'$  when  $\tau_b > 0$ . Since, the feasible set  $\mathcal{F}$  with relaxed lower bound includes the feasible set  $\mathcal{F}'$  with some positive lower bound, i.e.,

$$\mathcal{F}' \subseteq \mathcal{F} \quad (4.29)$$

the optimal value of the problem with relaxation on the lower bound must be always less than or equal to the optimal value of the problem with some set positive lower bound. In other words, the throughput computed with  $\tau_b > 0$ , can be no greater than the throughput computed with  $\tau_b = 0$ . ■

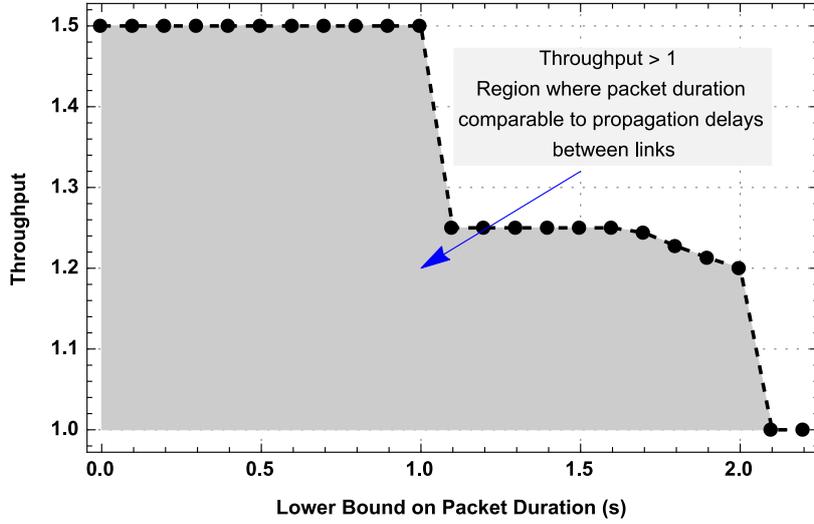


Figure 4.13: Isosceles triangle network geometry.

#### 4.7.1 Simulation results

We study the region in which the network must be operated in order to achieve the throughput gains due to the presence of large propagation delays. To study this, we consider two network geometries – a sea-trial network geometry shown in Fig. 4.2 and an isosceles triangle shaped network geometry. The propagation delay corresponding to the links are as shown in Table 4.9. We vary the lower bound ( $\tau_{lb}$ ) on packet duration and compute the schedules and the corresponding throughput for both the network geometries considered. The resulting throughput is plotted and shown in Fig. 4.12 and 4.13. We present the following observations and insights from the simulation results:

1. For both the network geometries, the highest throughput is achieved when  $\tau_{lb} = 0$  and supports the proposition. The value of throughput is 1.484 for the sea-trial network geometry and is 1.5 for the isosceles triangle shaped network geometry.  $\tau_{lb} = 0$  provides the maximum degree of freedom in the

TABLE 4.9: LINK PROPAGATION DELAYS

Link	Sea-Trial Network (s)	Isosceles Triangle (s)
(1, 2)	0.3890	1
(2, 1)	0.3890	1
(2, 3)	0.6130	2
(3, 2)	0.6130	2
(1, 3)	0.6052	2
(3, 1)	0.6052	2

variability of the packet duration and hence leads to maximum throughput. Theoretically, this might cause some of the packet lengths to be zero. However, the lower bound on the packet duration can be increased to a non-zero value based on the minimum duration that each node must transmit per frame.

2. Throughput is a non-increasing function of the lower bound on the packet duration. For example, it remains at 1.5 till the value of  $\tau_b = 1$ , in the case of isosceles shaped network geometry and then reduces further as the lower bound is increased.
3. For terrestrial wireless networks using radio-frequency waves, the highest throughput that can be achieved is not greater than 1. Both in Fig. 4.12 and Fig. 4.13, we see that the shaded region is where the throughput is higher than the throughput achievable in terrestrial wireless networks.
4. Throughput reduces to 1 for both the network geometries when,

$$\tau_b > \max_{\forall (j,k) \in \mathcal{L}} D_{jk}. \quad (4.30)$$

So, only when the packet durations are less than the maximum propagation

delay in the network, throughput gain is possible. In fact, this is the definition of the large propagation delay, since large propagation delays are relative to the transmission duration considered.

## 4.8 Summary

We formulated an optimization problem with the goal of finding throughput-maximizing schedules. We exploited large propagation delays by minimizing the fractional idle time in a frame, given the packet traffic demands on the links considered in a practical UWA network. In contrast to the approach where the minimum length schedules are computed by minimizing the frame length, we considered the fractional idle time to allow variability in the packet transmission duration. We presented and visualized the time-slotted fixed packet duration solution and compared it with the proposed MILFP solution resulting in schedules with no time slots and variable packet duration. The variability in the packet duration was crucial in achieving the throughput gain, thereby resulting in throughput closer to the upper bound  $\frac{N}{2}$ . The proposed algorithm outperformed the existing state-of-the-art methods to find schedules exploiting the large propagation delays. We computed schedules for some illustrative network geometries with arbitrary packet traffic demands and compared it with the previous work. The centralized algorithm proposed is used to compute schedules for multi-hop multi-line grid networks to demonstrate the scalability of the algorithm. We studied the operating region for UWA networks in which, throughput gains due to the large propagation delays can be achieved. We presented throughput behavior, as the lower bound on packet duration

is varied. This study provided insights and quantified the regions where the propagation delays and transmission duration are comparable. The results are presented to confirm the benefits of unslotted variable packet duration schedules in maximizing throughput for arbitrary UWA networks.

## Chapter 5

# Robust & Unslotted Schedules for Practical Multihop Grid Networks

---

In Chapter 4, unslotted schedules with variable packet lengths are computed which outperform the current state-of-the-art time-slotted and unslotted schedules. We considered in Section 4.6, the scalability analysis of the algorithm and presented the solutions for much larger multi-hop multi-line grid networks. In this chapter, we present an in-depth study of practical multi-hop multi-line grid networks in presence of propagation delay uncertainties. We systematically generalize the assumptions in [3] and propose an algorithm to compute throughput-maximizing schedules for practical grid networks which perform better than the current state-of-the-art. The work presented in this chapter is published in [101], [102].

When operating over a large area, the multi-line grid topology with multi-hop relaying can be considered for providing high-rate services [3]. Moreover, multi-hop transmissions in which a longer distance is divided into multiple shorter hops, offer favorable bandwidth and path loss conditions than longer hops with larger transmission power [51], [103]. Multi-line grid topology considered in [3] consists of parallel lines with regularly placed nodes. Messages originate from the first node on each line and are destined to the final node on the

same line. Intermediate nodes act as relay nodes which receive the incoming packets, decode them and retransmit them to the next hop, until they reach the final destination node. For multi-hop networks with half-duplex underwater acoustic modems acting as the relays and with the interference among hops due to the broadcast nature of the acoustic channel, characterizing the capacity of the network is an extremely difficult problem [104]. However, the throughput upper bound was established in [3] for a particular network setting with a regular  $N$ -node multi-line grid network. The spacing between neighboring nodes on the same line was considered to be one unit and the distance separating every two adjacent lines to be two units. The motivation for using such a multi-hop network and the features of grid topology are elaborated in [3]. Although, the study in [3] provides significant insights into exploiting the large propagation delay in regular multi-line grid network, there are a few assumptions which when generalized, lead to more complex and interesting problems, the solution of which cannot be directly obtained from the techniques presented in [3]. We consider two such problems:

- *Irregularity due to deployment:* The deployment of the nodes while maintaining the regularity of the grid networks is extremely difficult. It is not surprising to end up with a grid network which has the distance matrix close to the desired matrix, but not exactly the same after deployment. The assumptions on the regular spacing between the nodes will no longer be valid. This results in non-integer propagation delays among the nodes in contrast to the integer propagation delays considered in [3]. The

---

schedules can be computed by approximating the non-integer propagation delays to the nearest integers as proposed in [1]. However, due to this approximation, we have to allocate sufficient guard times at the start and end of the time slots to prevent early and delayed receptions within the time slots [1],[105]. The approximations are the cause for inefficient utilization of slots thereby reducing the throughput. The time-slotted nature of the solution constrains the transmissions to be strictly within the time slots. We propose unslotted transmission schedules for the practical grid networks considered by formulating the scheduling problem as a MILFP and compute schedules which do not require explicit transmission slots to transmit.

- *Propagation delay uncertainties:* Propagation delay is assumed to be known with absolute certainty in [3]. However, uncertainties in propagation delay between the nodes exist, arising from slight drift of nodes around the deployed locations due to varying ocean current, and uncertainty in the measurement of exact location of the underwater nodes. Moreover, propagation delay underwater is parameterized by temperature, salinity, depth, etc. [106]. Any changes in these physical parameters might also cause uncertainty in the assumed propagation delay values. We include these uncertainties in the model proposed and formulate a MILFP to compute schedules which are robust to such uncertainties.

We first demonstrate the throughput gains that can be achieved in grid networks using unslotted schedules and compare this to the state-of-the-art time-slotted

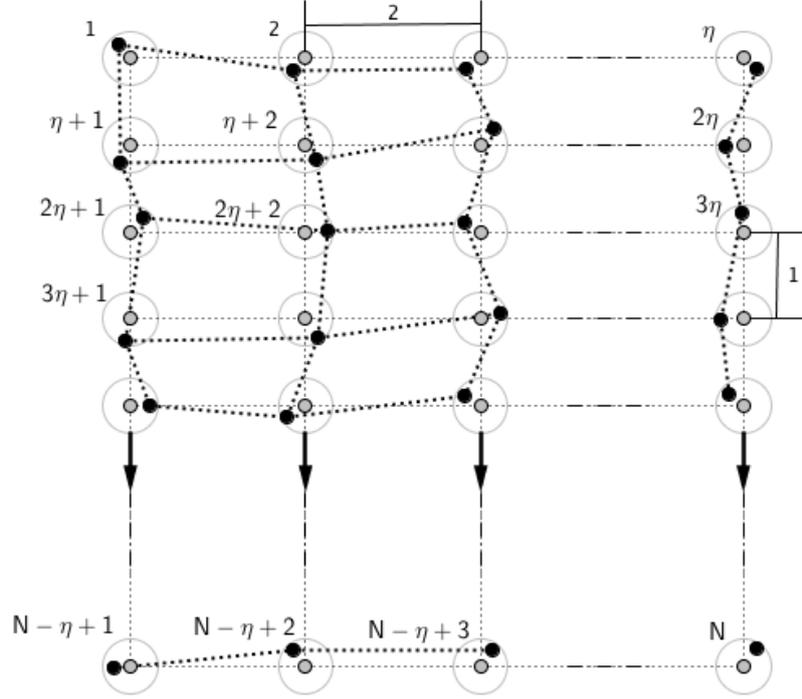


Figure 5.1: Practical  $N$ -node multi-line grid network with unevenly spaced nodes.

schedules presented in [3]. We derive the necessary condition for the proposed algorithm to always perform better than the time-slotted solution. Next, we demonstrate that proposed schedules are robust under propagation delay uncertainties.

### 5.1 System Model and Assumptions

The system model and assumptions are as follows:

1. We consider  $N$ -node multi-line grid network with multi-hop relaying and non-negligible propagation delay among the nodes (see Fig. 4.9 & 5.1). Each node  $i \in \{1, \dots, N\}$  is a half-duplex underwater acoustic modem, i.e., the node cannot receive and transmit simultaneously. All the transmissions are assumed to be unicast and intended to their corresponding destinations,

i.e., the node cannot receive from or transmit to more than one node simultaneously.

2. The number of independent node lines in the network is  $\eta \geq 1$ . Information-bearing data packets originate from the source nodes  $1, 2, \dots, \eta$  at one end of the line and are relayed hop by hop to the destination nodes  $N - \eta + 1, N - \eta + 2, \dots, N$  at the other end of the line as illustrated in Fig. 4.9 & 5.1.
3. Due to the architecture of the network and the information flow, the links considered for the transmission of the data packets are fixed, given  $N$  and  $\eta$ . We denote the set of these directed links by  $\mathcal{L}_s$  and the set of all possible links in the network by  $\mathcal{L}$ . Each link in the set  $\mathcal{L}$  or  $\mathcal{L}_s$  can be explicitly written as a 2-tuple  $(j, k)$  which represents a link where node  $j$  is the transmitter and node  $k$  is the receiver. The propagation delay corresponding to link  $(j, k)$  is  $D_{jk}$  units.
4. The nodes in the network are intended to be deployed such that all the links in a line have a unit propagation delay and the adjacent node lines are separated by two units of propagation delay (similar to [3]) as shown in Fig. 4.9. However, in actual deployment, the nodes are slightly displaced from the regular grid alignment (see Fig. 5.1). The propagation delay  $D_{jk}$  corresponds to the measured delay after the deployment of nodes, i.e.,  $D_{jk}$  is close to 1 unit but not exactly 1:

$$\lfloor D_{jk} \rfloor = 1 \quad \forall (j, k) \in \mathcal{L}_s \quad (5.1)$$

where  $\lfloor \cdot \rfloor$  denotes the operation to round off the value to the closest integer.

5. We also consider the uncertainty in the propagation delay due to reasons such as node mobility because of ocean currents, uncertainty in the measurement of exact deployed locations and physical changes in the ocean environment. We model the uncertainty or variation in the propagation delay by assuming that they are known to lie in a given set. Specifically, the uncertain propagation delay denoted by  $\hat{D}_{jk}$  for each link  $(j, k) \in \mathcal{L}$  is associated with an uncertainty set denoted by  $\mathcal{D}_{jk}$ , i.e.,  $\hat{D}_{jk} \in \mathcal{D}_{jk} \forall (j, k) \in \mathcal{L}$ .
6. To characterize the uncertainty set  $\mathcal{D}_{jk}$ , we formulate the uncertainty in the deployed locations to lie in a Euclidean ball [107] defined as:

$$B(\mathbf{x}_i, r_i) = \{\mathbf{x}_i + r_i \mathbf{u} \mid \|\mathbf{u}\|_2 \leq 1\}, \quad i \in 1, \dots, N. \quad (5.2)$$

The vector  $\mathbf{x}_i$  is the center of the ball denoting the expected location of deployment of node  $i$  and the scalar  $r_i$  is the radius of the ball as illustrated in Fig. 5.1. We assume that the expected locations  $\mathbf{x}_i$  and radius  $r_i$  are known and the node  $i$  deployed can lie at any location inside  $B(\mathbf{x}_i, r_i)$ . All possible propagation delays that can be computed between the points lying in  $B(\mathbf{x}_j, r_j)$  and  $B(\mathbf{x}_k, r_k)$  form the set  $\mathcal{D}_{jk}$ .

7. We partition time into frames, but do not further partition into slots as is commonly done. The time, relative to the start of the frame, at which node  $j$  starts transmitting a packet to node  $k$ , is  $t_{jk}$ . The packet/transmission

duration corresponding to the packet transmission on link  $(j, k)$  is  $\tau_{jk}$ .

The frame length of the schedule (also termed as period of the schedule in [1],[3]) is  $T$ .

8. The schedules are computed based on the available information on the network topology while exploiting the large propagation delay between the nodes. We consider two different problems as listed below:

- In the first case, we consider the problem of finding optimal transmission schedules utilizing the propagation delay information assuming that the nodes have been deployed and the location information is accurately known.
- In the second case, we consider the problem of finding the robust optimal schedules for the considered multi-line grid network, in the presence of uncertainty in the propagation delay attributed to various reasons like inaccurately positioned nodes or drift in location of nodes due to ocean waves, changes in the ocean's physical parameters, etc.

9. We adopt a protocol channel model [87] and denote by  $\alpha$ , the ratio of interference range to the communication range as already described in Section 3.1.

## 5.2 Time-Slotted $\rho$ -Schedules for Multihop Networks

The schedules computed utilizing the long propagation delay in UWA networks using algorithms presented in [1],[3] are time-slotted. Moreover, these algorithms only accept integer propagation delays. The  $\rho$ -Schedule was first defined in [1]

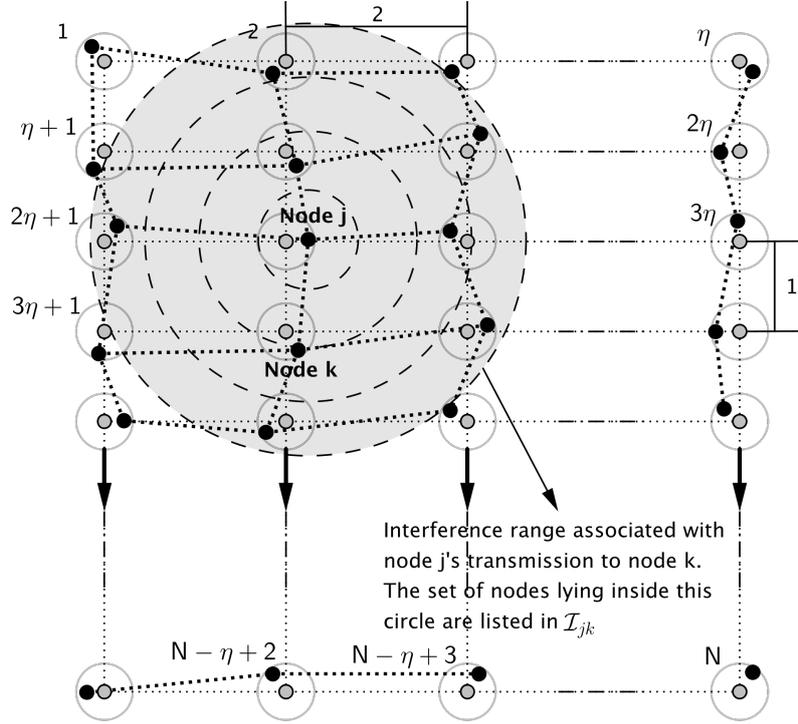


Figure 5.2: The figure shows the nodes lying in the collision domain associated with node  $j$ 's transmission to node  $k$ . The list of these nodes are enumerated in the set  $\mathcal{I}_{jk}$ . Note that due to the slight irregularity in the grid network, the number of nodes lying in the interference range associated with each link  $(j, k)$  can be different. This is not the case in regular grid networks considered in [3].

for a network with non-integer propagation delays. The network geometry is represented in the form of a delay matrix  $\mathbf{D}$  as defined in (2.1), where each element of the delay matrix contains the propagation delay between the corresponding transmitter-receiver pair. For a non-integer delay matrix  $\mathbf{D}$ , the elements can be rounded off to yield an integer delay matrix  $\mathbf{D}'$ . The approximated integer delay matrix  $\mathbf{D}'$  is then used to compute the time-slotted  $\rho$ -Schedules. However, the  $\rho$ -Schedule presented in [1] assumes a single collision domain network whereas here we consider a multi-hop network with multiple partially overlapping collision domains.

Each transmitter node  $j$  on link  $(j, k)$  is associated with its collision domain. There are  $N - \eta$  partially overlapping collision domains (also illustrated in [3]) due to  $N - \eta$  transmitters. For each link  $(j, k) \in \mathcal{L}_s$ , we enumerate all the nodes in its corresponding collision domain and denote the set of these nodes by  $\mathcal{I}_{jk}$  (see Fig. 5.2). Hence,  $\mathcal{I}_{jk}$  contains a list of nodes which lie in the interference range corresponding to node  $j$ 's transmission to node  $k$ . Given the set,  $\mathcal{I}_{jk} \forall (j, k) \in \mathcal{L}_s$ , the values of  $\rho^+$  and  $\rho^-$  are computed as follows:

$$\rho^+ = \max_{\forall (j,i)} (D_{ji} - D'_{ji}) \quad \forall i \in \mathcal{I}_{jk} \text{ and } \forall (j, k) \in \mathcal{L}_s \quad (5.3)$$

$$\rho^- = - \min_{\forall (j,i)} (D_{ji} - D'_{ji}) \quad \forall i \in \mathcal{I}_{jk} \text{ and } \forall (j, k) \in \mathcal{L}_s \quad (5.4)$$

where  $\rho^+$  and  $\rho^-$  are the largest approximations made in the non-integer propagation delays. Since the schedules are time-slotted, let us denote the time slot length by  $\tau$ . The duration for which the transmission slot is utilized for transmission in a  $\rho$ -Schedule was derived in [1], and is given by:

$$t_{\text{pd}} = \tau(1 - \rho^+ - \rho^-). \quad (5.5)$$

$t_s = \tau\rho^-$  and  $t_e = \tau\rho^+$  define the guard times allocated at the start and end of the time slots to prevent collisions due to the approximations made in the propagation delays.

The throughput of a regular multi-line grid  $N$ -node network with multi-hop relaying is upper bounded by  $(N - \eta)/2$  [3], Th. 2, and there exists schedules which when adopted achieve this throughput. We can define, the throughput  $S_\rho$

of a  $\rho$ -Schedule as:

$$S_\rho = \left(\frac{N - \eta}{2}\right) \left(\frac{t_{\text{pd}}}{\tau}\right) = \frac{N - \eta}{2} (1 - \rho^+ - \rho^-). \quad (5.6)$$

The throughput  $S_\rho$  is the number of successful transmissions per time slot multiplied by the utilization efficiency in a time slot. Therefore, the values of  $\rho^+$  and  $\rho^-$  play a critical role in determining the achievable throughput while using  $\rho$ -Schedules. The optimal time slot lengths which minimize the guard times were computed in [105] for a single collision domain network. In the next section, we formulate the scheduling problem to compute unslotted schedules which do not require explicit transmission slots.

### 5.3 Problem Formulation

Note that part of the formulation presented in this section is similar to the problem formulated in Section 4.2. The only difference being one transmission per frame considered in the case of multi-hop grid networks and the inclusion of multiple partially overlapping collision domains. Despite the similarities with the earlier formulation; to maintain the continuity, in this section, we present the formulation with the differences mentioned.

Consider a pair of links  $(j, k), (l, i) \in \mathcal{L}_s$  such that node  $j$  starts transmitting a packet to node  $k$  at time  $t_{jk}$  with packet duration  $\tau_{jk}$ , and node  $l$  transmits a packet to node  $i$  at time  $t_{li}$  with packet duration  $\tau_{li}$ . Collision at receiver node  $i$  occurs if its desired message from transmitter node  $l$  overlaps with undesired message from node  $j$ 's transmission to node  $k$ . Note that for node  $j$ 's transmission not to interfere with node  $i$ 's reception, either of the following

conditions need to be satisfied:

$$t_{jk} + \tau_{jk} + D_{ji} \leq t_{li} + D_{li} \quad (5.7)$$

(or)

$$t_{jk} + D_{ji} \geq t_{li} + \tau_{li} + D_{li} \quad (5.8)$$

$\forall \{(j, k), (l, i) \in \mathcal{L}_s | D_{ji} \leq \alpha D_{jk}\}$ , where the condition  $D_{ji} \leq \alpha D_{jk}$  is satisfied when node  $i$  lies in the interference range of node  $j$ . Note that, a collision domain can be identified with each transmitting node in the network. Since there are  $N - \eta$  links to be scheduled in the set  $\mathcal{L}_s$ , there are  $N - \eta$  partially overlapping collision domains associated with each transmitter. The message is considered as an interference at all other nodes (in the collision domain) except for the destination node. The transmission start times  $t_{jk}, t_{li}$  and corresponding packet duration  $\tau_{jk}, \tau_{li}$  must be chosen such that the desired message at node  $i$  is interference-free.

If the transmission scheduling decisions are taken considering the propagation delay based constraints listed in (5.7) and (5.8), the corresponding receptions at the nodes will be interference-free, i.e., there will be no collisions within a frame. However, note that the following possibilities are not captured in the constraints listed in (5.7) and (5.8):

- The transmissions from the previous frames interfering with reception in the current frame.
- The transmission in the current frame interfering with possible reception

in the subsequent frames.

Ignoring the above conditions will result in poor schedules. In order to consider the inter-frame constraints, we generalize the inequalities:

$$t_{jk} + \beta T + \tau_{jk} + D_{ji} \leq t_{li} + D_{li} \quad (5.9)$$

(or)

$$t_{jk} + \beta T + D_{ji} \geq t_{li} + \tau_{li} + D_{li} \quad (5.10)$$

$\forall \{(j, k), (l, i) \in \mathcal{L}_s | D_{ji} \leq \alpha D_{jk}\}$ , where  $\beta \in \mathbb{Z}$  is the integer constant which determines the number of adjacent frames in the past and future, that are taken into account. Note that setting  $\beta = 0$ , in (5.9) and (5.10) leads to (5.7) and (5.8) respectively.  $\beta = -1$ , corresponds to conditions considering one adjacent frame in the past while  $\beta = 1$ , corresponds to conditions considering one adjacent frame in the future.

### 5.3.1 Propagation delay constraints

The conditions listed in (5.9) and (5.10) form a set of disjunctive constraints which result in the feasible set forming a non-convex region over which the search for the solution is required. There are two well known methods for conversion of disjunctive constraints to conjunctive constraints: convex hull reformulation [94] and Big-M reformulation [95], [96]. We use the Big-M transformation to convert the disjunctive constraints (5.9) & (5.10) into conjunction and rearrange them as follows:

1.  $\beta = 0$  :

$$t_{jk} - t_{li} + \tau_{jk} \leq -(D_{ji} - D_{li}) + Mp_{jk,li} \quad (5.11)$$

$$-t_{jk} + t_{li} + \tau_{li} \leq (D_{ji} - D_{li}) + M(1 - p_{jk,li}) \quad (5.12)$$

2.  $\beta = 1$  :

$$t_{jk} - t_{li} + \tau_{jk} + T \leq -(D_{ji} - D_{li}) + Mq_{jk,li} \quad (5.13)$$

$$-t_{jk} + t_{li} + \tau_{li} - T \leq (D_{ji} - D_{li}) + M(1 - q_{jk,li}) \quad (5.14)$$

3.  $\beta = -1$  :

$$t_{jk} - t_{li} + \tau_{jk} - T \leq -(D_{ji} - D_{li}) + Mr_{jk,li} \quad (5.15)$$

$$-t_{jk} + t_{li} + \tau_{li} + T \leq (D_{ji} - D_{li}) + M(1 - r_{jk,li}) \quad (5.16)$$

where  $p_{jk,li}$ ,  $q_{jk,li}$  and  $r_{jk,li}$  are the binary variables<sup>1</sup> associated with each pair of disjunctive constraints considered  $\forall\{(j, k), (l, i) \in \mathcal{L}_s | D_{ji} \leq \alpha D_{jk}\}$ . The inequalities (5.9) and (5.10) are presented in (5.11), (5.12), (5.13), (5.14), (5.15) and (5.16) after the Big-M transformation for the values of  $\beta$  from the set  $\{-1, 0, 1\}$ .

---

<sup>1</sup>With the binary variable taking value 0 or 1 (e.g.  $p_{jk,li} = 0$  or 1) along with a large enough value of parameter M, one of the constraints in the disjunctive pair becomes redundant. Note that smaller the value of M is, the tighter the Big-M reformulation can be. We select an arbitrarily large value of M for the transformation.

### 5.3.2 Allowing transmissions and receptions across frame boundary

A packet transmission in the current frame on link  $(j, k)$  at time  $t_{jk}$ , can cause interference till time  $t_{jk} + \alpha D_{jk} + \tau_{jk}$ . To prevent the end of the packet transmission on the link from causing maximum interference to cross the subsequent frame, we impose the following constraint:

$$t_{jk} + \alpha D_{jk} + \tau_{jk} < 2T. \quad (5.17)$$

Note that the above constraint does not restrict the transmissions in the frame to be fully contained within the frame.

### 5.3.3 Throughput

The average throughput  $S$  of a schedule with frame length  $T$  can be computed by summing the total reception time (or equivalently, the total transmission time) on all nodes in the network in one frame duration  $T$ :

$$S = \frac{\sum_{(j,k) \in \mathcal{L}_s} \tau_{jk}}{T}. \quad (5.18)$$

### 5.3.4 Objective function

#### Minimizing frame length

In [2], the objective considered was to minimize the frame length  $T$  while keeping the packet duration fixed. Even if the packet duration is variable, minimizing  $T$  results in the solution in which packet duration is equal in size across all transmissions, and moreover is equal to the lower bound set on it [86].

Minimizing  $T$  prevents the full exploitation of large propagation delays in UWA networks. To utilize the degree of freedom that is provided by varying the packet duration, we formulate a different objective which results in variable packet duration [86]. Moreover, minimizing  $T$  either requires the packet duration to be set a constant, or else, if the packet duration is variable, then it must be assigned a positive lower bound value equal to the desired packet duration. This is not desirable since, we do not know the value of packet duration necessary for optimal network throughput. Therefore, minimizing the frame length is not a desirable objective.

#### Minimizing idle time

Next, consider the objective function to be minimized as the total idle time in a frame given by

$$f_{\text{MIT}} = \frac{(N - \eta)T - 2 \left[ \sum_{\forall(j,k) \in \mathcal{L}_s} \tau_{jk} \right]}{T}. \quad (5.19)$$

The first term in (5.19) is the total sum of time available for the scheduling per frame. The second term consists of the amount of time the nodes are busy with transmissions and receptions. Although minimizing the idle time results in better exploitation of long propagation delays due to variability in the packet duration, the variability itself can cause unfair distribution of the time among the transmitting nodes. The unfair distribution of time is not desirable for the multi-line grid networks. A link with less time allocated for transmission will act as a bottleneck for the information flow and thereby reduce the achievable throughput. We require the packet duration to be all of equal size and at the same time be able to exploit the large propagation delay to maximize the

throughput.

### Maximizing the minimum packet duration

In order to maximize the throughput and at the same time maintain the fairness among the nodes considered for scheduling, we consider the objective of maximizing the minimum packet duration. Let the minimum packet duration be denoted by  $z$ , i.e.,

$$z = \min\{\tau_{jk} | (j, k) \in \mathcal{L}_s\}. \quad (5.20)$$

Note that maximizing  $z$ , subject to condition  $\tau_{jk} \geq z \forall (j, k) \in \mathcal{L}_s$ , along with the propagation delay constraints provide solutions in which the packet duration in the solution is unbounded above since  $z \in \mathbb{R}^+$ . To prevent this, we consider minimizing an objective similar to fractional idle time as follows:

$$f_{\text{MMPD}} = \frac{(N - \eta)T - 2(N - \eta)z}{T}. \quad (5.21)$$

This objective minimizes the frame length  $T$ , while maximizing the minimum packet duration  $z$ . Based on the formalized objective and the constraints listed,

the following Mixed-Integer Linear Fractional Problem (MILFP) is setup:

$$\begin{aligned}
& \min && f_{\text{MMPD}} \\
& \text{s.t.} && t_{jk} - t_{li} + \tau_{jk} - Mp_{jk,li} \leq -(D_{ji} - D_{li}) \\
& && -t_{jk} + t_{li} + \tau_{li} + Mp_{jk,li} \leq (D_{ji} - D_{li}) + M \\
& && t_{jk} - t_{li} + \tau_{jk} + T - Mq_{jk,li} \leq -(D_{ji} - D_{li}) \\
& && -t_{jk} + t_{li} + \tau_{li} - T + Mq_{jk,li} \leq (D_{ji} - D_{li}) + M \tag{5.22} \\
& && t_{jk} - t_{li} + \tau_{jk} - T - Mr_{jk,li} \leq -(D_{ji} - D_{li}) \\
& && -t_{jk} + t_{li} + \tau_{li} + T + Mr_{jk,li} \leq (D_{ji} - D_{li}) + M \\
& && t_{jk} + \tau_{jk} - 2T < -\alpha D_{jk} \\
& && -\tau_{jk} + z \leq 0
\end{aligned}$$

**Proposition 1**

The schedules computed by solving MILFP in (5.22) result in the throughput  $S$  greater than the throughput  $S_\rho$  computed using  $\rho$ -Schedule, if and only if  $(1 - \rho^+ - \rho^-)T \leq 2z$ .

**Proof**

To ensure  $S \geq S_\rho$ , i.e., the throughput obtained by using  $\rho$ -Schedule is less than the throughput computed using MILFP schedule, the inequality:

$$\left(\frac{N - \eta}{2}\right) \frac{t_{\text{pd}}}{\tau} \leq \frac{\left(\sum_{(j,k) \in \mathcal{L}_s} \tau_{jk}\right)}{T} \tag{5.23}$$

needs to be satisfied (using (5.6) and (5.18)). The minimum value of R.H.S. in (5.23) can be obtained by considering the minimum values of the packet

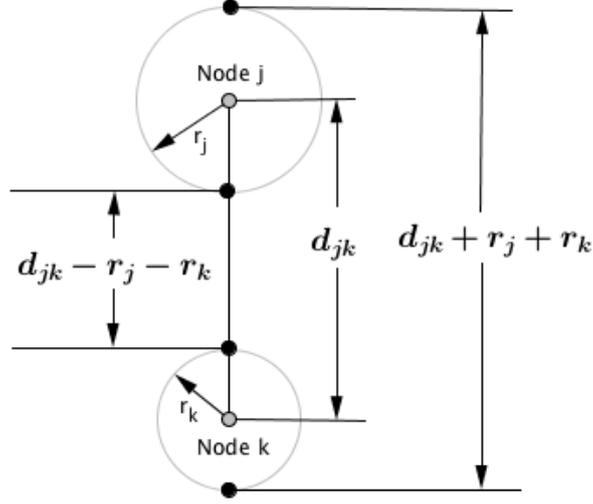


Figure 5.3: Propagation delay uncertainty set considering the worst case scenarios.

duration  $\tau_{jk}$ . Since we aim to maximize the minimum packet duration  $z$ , the minimum value of the sum of packet duration occurs when  $\tau_{jk} = z \forall (j, k) \in \mathcal{L}_s$ .

Therefore,

$$\frac{\left( \sum_{(j,k) \in \mathcal{L}_s} \tau_{jk} \right)}{T} = \frac{(N - \eta)z}{T}. \quad (5.24)$$

Therefore, (5.23) can now be written as :

$$\left( \frac{N - \eta}{2} \right) (1 - \rho^+ - \rho^-) \leq \frac{(N - \eta)z}{T} \quad (5.25)$$

$\Rightarrow$

$$(1 - \rho^+ - \rho^-)T \leq 2z. \quad (5.26)$$

■

### 5.3.5 Robust MILFP formulation

The scheduling problem in (5.22) is formulated with the assumption that the deployed locations are known with certainty and that once the nodes are deployed, the propagation delay between the nodes is accurately known. In reality, there are various sources of uncertainty in the measurement of propagation delay among the nodes. Next, we present the problem formulation which addresses such uncertainties.

The uncertainties in the propagation delay for the link  $(j, k)$  can be characterized by defining the set  $\mathcal{D}_{jk}$  from which  $\hat{D}_{jk}$  can take values as discussed in Section 5.1. If  $d_{jk}$  is the distance between node  $j$  and node  $k$  then Fig. 5.3 shows the distances between node  $j$  and node  $k$  in the worst cases. Assuming that the nodes only lie inside the ball of corresponding radius  $r_j$  and  $r_k$ , the bounds on  $\hat{D}_{jk}$  are as follows:

$$\hat{D}_{jk} \in \left[ \frac{d_{jk} - r_j - r_k}{c}, \frac{d_{jk} + r_j + r_k}{c} \right] \forall (j, k) \in \mathcal{L}. \quad (5.28)$$

In the presence of these uncertainties, we require the constraints in (5.22) to be satisfied for all possible values of the parameter  $\hat{D}_{jk} \in \mathcal{D}_{jk}$  for the solution to be robust [107]. A sub-problem is set up to be solved for each constraint with uncertain parameter:

$$\min_{\hat{D}_{ji} \in \mathcal{D}_{ji}, \hat{D}_{li} \in \mathcal{D}_{li}} \{ -(\hat{D}_{ji} - \hat{D}_{li}) \}. \quad (5.29)$$

Note that (5.29) is an unconstrained linear optimization problem with known

bounds on the values of each variable  $\hat{D}_{ji}$  and  $\hat{D}_{li}$ , given by:

$$\frac{d_{ji} - r_j - r_i}{c} \leq \hat{D}_{ji} \leq \frac{d_{ji} + r_j + r_i}{c} \quad (5.30)$$

$$\frac{d_{li} - r_l - r_i}{c} \leq \hat{D}_{li} \leq \frac{d_{li} + r_l + r_i}{c}. \quad (5.31)$$

Using (5.30) and (5.31), we solve the following sub-problems for the robust linear constraints in (5.35):

$$\min_{\hat{D}_{ji} \in \mathcal{D}_{ji}, \hat{D}_{li} \in \mathcal{D}_{li}} \{-(\hat{D}_{ji} - \hat{D}_{li})\} = -(D_{ji} - D_{li} + \frac{r_j + r_l + 2r_i}{c}) \quad (5.32)$$

$$\min_{\hat{D}_{ji} \in \mathcal{D}_{ji}, \hat{D}_{li} \in \mathcal{D}_{li}} \{(\hat{D}_{ji} - \hat{D}_{li})\} = D_{ji} - D_{li} - \frac{r_j + r_l + 2r_i}{c} \quad (5.33)$$

and

$$\min_{\hat{D}_{jk} \in \mathcal{D}_{jk}} \{\hat{D}_{jk}\} = D_{jk} - \frac{2(r_j + r_k)}{c}. \quad (5.34)$$

The R.H.S. in the constraints with uncertain parameters in (5.35) can now be

replaced with the solution to the sub-problems listed in (5.32), (5.33) and (5.34).

$$\begin{aligned}
 & \min && f_{\text{MMPD}} \\
 \text{s.t.} &&& t_{jk} - t_{li} + \tau_{jk} - Mp_{jk,li} \leq \min_{\hat{D}_{ji} \in \mathcal{D}_{ji}, \hat{D}_{li} \in \mathcal{D}_{li}} \{-(\hat{D}_{ji} - \hat{D}_{li})\} \\
 &&& -t_{jk} + t_{li} + \tau_{li} + Mp_{jk,li} \leq \min_{\hat{D}_{ji} \in \mathcal{D}_{ji}, \hat{D}_{li} \in \mathcal{D}_{li}} \{(\hat{D}_{ji} - \hat{D}_{li}) + M\} \\
 &&& t_{jk} - t_{li} + \tau_{jk} + T - Mq_{jk,li} \leq \min_{\hat{D}_{ji} \in \mathcal{D}_{ji}, \hat{D}_{li} \in \mathcal{D}_{li}} \{-(\hat{D}_{ji} - \hat{D}_{li})\} \\
 &&& -t_{jk} + t_{li} + \tau_{li} - T + Mq_{jk,li} \leq \min_{\hat{D}_{ji} \in \mathcal{D}_{ji}, \hat{D}_{li} \in \mathcal{D}_{li}} \{(\hat{D}_{ji} - \hat{D}_{li}) + M\} \\
 &&& t_{jk} - t_{li} + \tau_{jk} - T - Mr_{jk,li} \leq \min_{\hat{D}_{ji} \in \mathcal{D}_{ji}, \hat{D}_{li} \in \mathcal{D}_{li}} \{-(\hat{D}_{ji} - \hat{D}_{li})\} \\
 &&& -t_{jk} + t_{li} + \tau_{li} + T + Mr_{jk,li} \leq \min_{\hat{D}_{ji} \in \mathcal{D}_{ji}, \hat{D}_{li} \in \mathcal{D}_{li}} \{(\hat{D}_{ji} - \hat{D}_{li}) + M\} \\
 &&& t_{jk} + \tau_{jk} - 2T < -\alpha \min_{\hat{D}_{jk} \in \mathcal{D}_{jk}} \{\hat{D}_{jk}\} \\
 &&& -\tau_{jk} + z \leq 0
 \end{aligned} \tag{5.35}$$

We show in Section 5.4 that the schedules computed by solving (5.35) are indeed robust to the uncertainties in the propagation delays.

## 5.4 Results

We demonstrate the merits of the proposed techniques by comparing the unslotted and robust schedules as described here:

- *Unslotted schedules* – In this case, the unslotted schedule is computed using (5.22) for a practical grid network deployed with no uncertainty in the propagation delay values. We fix the coordinates of the deployed nodes in the simulation. We compare the performance analytically with the time-slotted  $\rho$ -Schedules and verify the computed schedules in the simulator.

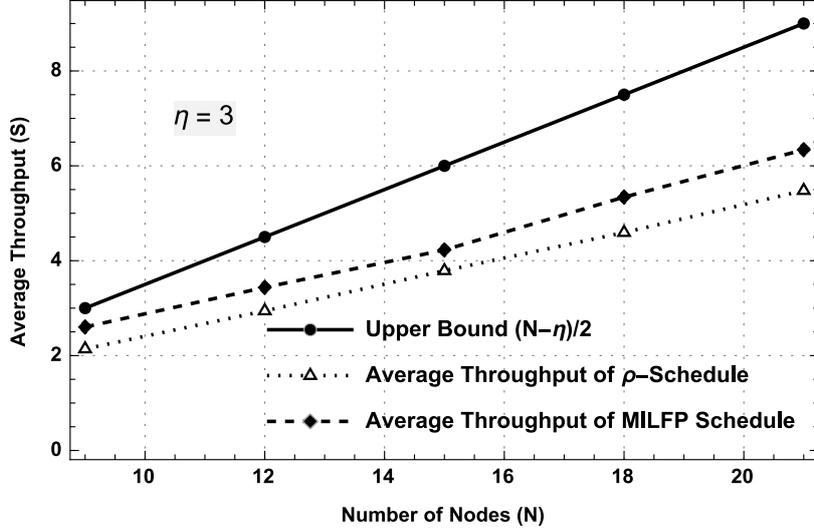


Figure 5.4: Average throughput is computed over 100 random instances of grid networks for different number of nodes.

- *Robust unslotted schedules* – For this case, the robust unslotted schedule is computed using (5.35) for a practical grid network deployed with a particular set of propagation delays and verify the same schedule for different sets of propagation delays by randomly perturbing the node locations within the allowed limit to demonstrate the robustness to the uncertainty in propagation delays.

We implement the computed schedules in the UNET simulator [108], [109] to verify the performance in various practical deployments of grid networks.

#### 5.4.1 Unslotted schedules

##### Throughput gain

To show that the schedules computed for the practical multi-hop multi-line grid networks using MILFP (5.22) perform better, we compute the  $\rho$ -Schedules and calculate the corresponding throughput and compare it with throughput

achieved using unslotted schedules. We generate 100 random instances of the grid networks with the different number of nodes  $N$  and fixed  $\eta = 3$ . We vary the number of nodes  $N$  from 9 to 21. For each random instance of the grid network, the  $\rho$ -Schedule and its corresponding throughput is computed using (5.6). The MILFP schedule is computed using (5.22) and the corresponding throughput is computed using (5.18). The average throughput is calculated over 100 random instances and plotted in Fig. 5.4 for each  $N$ . The MILFP schedules outperform the  $\rho$ -Schedules for all the network instances generated for different number of nodes as shown in Fig. 5.4 and also supports the Proposition 1, i.e.,

$$S_\rho \leq S \leq \frac{N - \eta}{2}. \quad (5.36)$$

The throughput gain is attributed to no time slotting requirement in the case of MILFP schedules, which reduces the amount of idle time and hence improves the time utilization efficiency in a frame.

#### **Throughput as a function of $\alpha$**

Nodes in the adjacent lines may or may not lie in the interference range of a transmission on a particular link. To increase the number of nodes lying in the interference range of a transmission, we increase  $\alpha$ . With a larger number of nodes in the interference range, the simultaneous transmission opportunities will reduce resulting in lower throughput. We vary the value of  $\alpha$  from 2 to 3 and compute the  $\rho$ -Schedules and MILFP schedules for each value of  $\alpha$  for a particular 12-node 3-line grid network. It is observed that the throughput as a function of  $\alpha$  is a non-increasing function as expected (see Fig. 5.5). As the

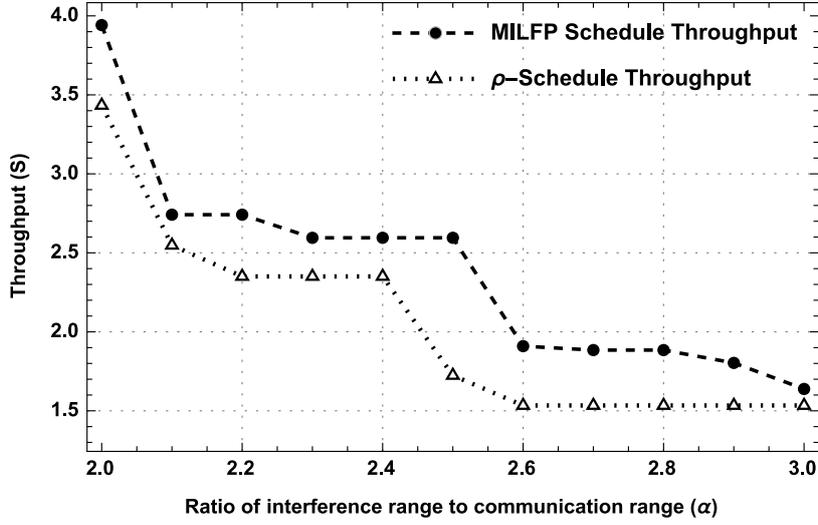


Figure 5.5: Throughput is plotted as the interference range increases causing more nodes in the adjacent lines to interfere with the transmissions.

interference range increases, the throughput reduces irrespective of the technique used to compute schedules. However, the throughput computed using unslotted schedules still remain higher than the throughput computed using  $\rho$ -Schedules as shown in Fig. 5.5.

### Throughput as a function of irregularity

We study the performance gap between unslotted schedules and  $\rho$ -Schedules as the irregularity in the network increases. The node  $i$  lies within a Euclidean ball of radius  $r_i$  centered at the deployed location. Let  $r_i = r, \forall i \in \{1, \dots, N\}$ , i.e., we set the radius corresponding to the Euclidean ball at each node in the grid network to the same value  $r$ . Note that the location of node  $i$  is chosen randomly to lie in this Euclidean ball  $B(\mathbf{x}_i, r)$ . Therefore, by increasing the radius  $r$ , the irregularity in the grid network deployed increases. For each value of  $r$ , 200 random instances of 12-node 3-line grid networks are generated and the

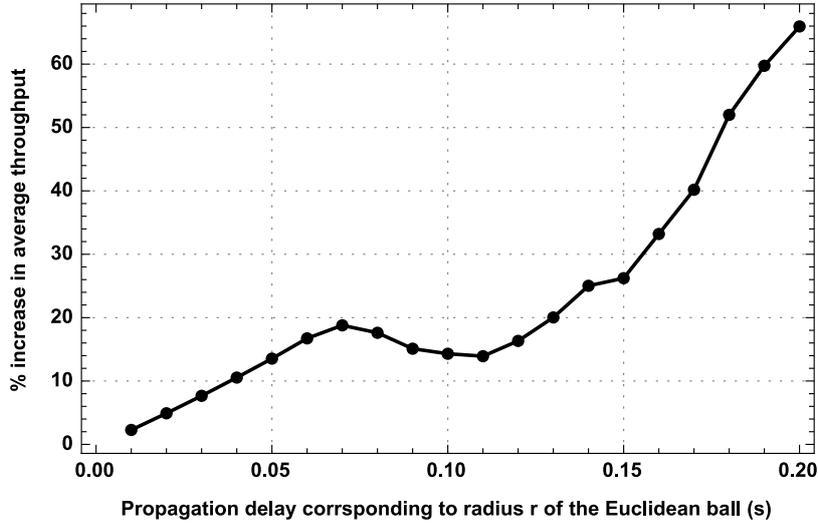


Figure 5.6: Average throughput is computed over 200 random deployments of 12-node 3-line multi-hop grid network. The irregularity in the network is introduced by varying the radius of the Euclidean ball in which the nodes are randomly deployed.

corresponding schedule and throughput is computed. The average throughput is computed over 200 random instances for each value of  $r$  and is plotted against the propagation delay corresponding to the value of  $r$  as shown in Fig. 5.6. The percentage increase is computed as  $\frac{S^{\text{avg}} - S_{\rho}^{\text{avg}}}{S_{\rho}^{\text{avg}}} \times 100$ , where  $S^{\text{avg}}$  is the average throughput computed using unslotted schedules and  $S_{\rho}^{\text{avg}}$  is the average throughput computed using  $\rho$ -Schedules. Fig. 5.7 shows the confidence intervals plotted as a notched box plot demonstrating the difference in medians of the throughput values of time-slotted and unslotted strategies. It is clear from the Fig. 5.6 & 5.7 that the performance of time-slotted  $\rho$ -Schedules decreases faster as the irregularity in the network increases than the performance of the unslotted schedules. In the case of  $\rho$ -Schedules, the time-slotted schedule is adopted and the guard times are introduced in the slots to mitigate the effect of irregularity in the network. It uses the same schedule while increasing the

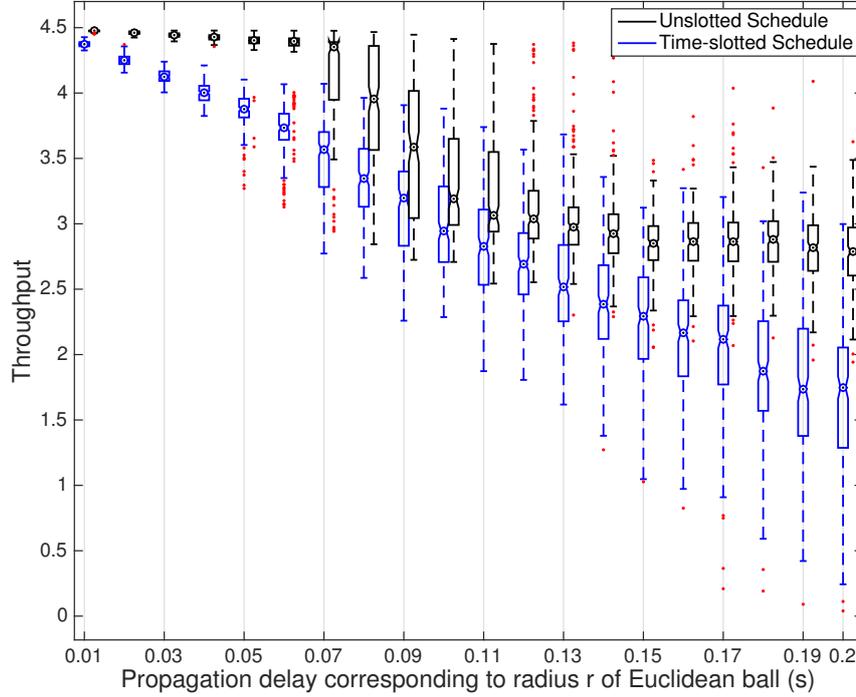


Figure 5.7: Throughput computed for each of the 200 random deployments at each of the selected radius  $r$  is shown as a notched box plot to demonstrate that the true medians do differ from each other with 95 % confidence.

guard times. However, in the case of the unslotted schedule computed using the MILFP algorithm, each time the network geometry becomes more irregular, the transmission schedule changes and the idle/guard times in the frame are minimized. This is the reason why the decrease in performance of unslotted schedules is lesser than that of time-slotted schedules.

### Verification in the simulator

In the simulator, we deploy the nodes to form a grid network. We consider a 12-node 3-line multi-hop grid network, i.e.,  $N = 12$  and  $\eta = 3$ . We set the sound speed  $c = 1540$  m/s. The coordinates for the 12 communication nodes are

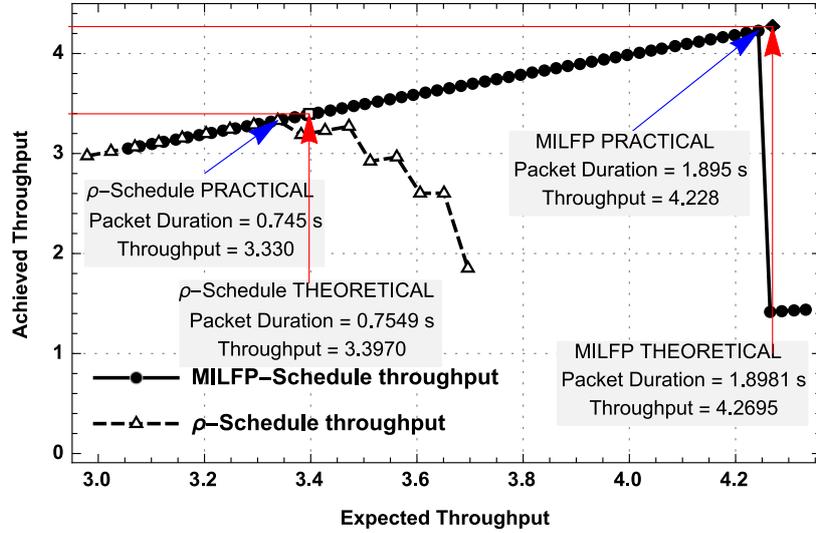


Figure 5.8: Verification of unslotted and time-slotted schedules on UNET simulator for 12-node 3-line mult-i-hop grid network.

TABLE 5.1:  $\rho$ -SCHEDULE

	Time slot 1	Time slot 2	Time slot 3	Time slot 4
<b>Node 1</b>	TX to 4	TX to 4	Idle	Idle
<b>Node 2</b>	Idle	Idle	TX to 5	TX to 5
<b>Node 3</b>	TX to 6	TX to 6	Idle	Idle
<b>Node 4</b>	TX to 7	RX from 1	RX from 1	TX to 7
<b>Node 5</b>	RX from 2	TX to 8	TX to 8	RX from 2
<b>Node 6</b>	TX to 9	RX from 3	RX from 3	TX to 9
<b>Node 7</b>	RX from 4	RX from 4	TX to 10	TX to 10
<b>Node 8</b>	TX to 11	TX to 11	RX from 5	RX from 5
<b>Node 9</b>	RX from 6	RX from 6	TX to 12	TX to 12
<b>Node 10</b>	RX from 7	Idle	Idle	RX from 7
<b>Node 11</b>	Idle	RX from 8	RX from 8	Idle
<b>Node 12</b>	RX from 9	Idle	Idle	RX from 9

chosen to form a perfect grid network as shown in Fig. 4.9 with 1 s propagation delay between adjacent nodes in a line and 2 s propagation delay between nodes in the adjacent lines. We set the communication range  $R_c = 1540$  m and the

TABLE 5.2: UNSLOTTED SCHEDULE

Link	Transmission Start Time (s)	Packet Duration (s)
(1, 4)	1.2008	1.8981
(2, 5)	3.0288	1.8981
(3, 6)	1.0058	1.8981
(4, 7)	3.9675	1.8981
(5, 8)	1.9670	1.8981
(6, 9)	0	1.8981
(7, 10)	3.0942	1.8981
(8, 11)	1.1116	1.8981
(9, 12)	2.8904	1.8981

interference range  $R_i = 2 \times 1540 = 3080$  m. However, note that we need to consider practical grid networks, in which there exist slight irregularity due to deployment. To achieve this, we introduce random perturbations in the coordinates of each of the 12 nodes of the perfect grid network to represent various random deployment locations. We ensure that each node  $i$  lie within the euclidean ball  $B(\mathbf{x}_i, r)$  of radius  $r = 154$  m. For each transmission on a link  $(j, k)$  with distance  $d_{jk}$ , its interference extends up to a distance  $2d_{jk}$ . In this simulation, the physical layer offered by a modem is replaced by a simulation model that mimics the behavior of the modem in the considered channel. We use the `HalfDuplexModem` model [108] which provides a generic implementation of a half-duplex modem that can be used to model many types of underwater modems.

For a particular random instance of 12-node 3-line multi-hop grid network, the  $\rho$ -Schedule and MILFP schedule are computed:

- *Computing  $\rho$ -Schedule*: The time-slotted schedule computed for a 12-node

3-line multi-hop grid network is shown in Table 5.1. The schedule can also be found in [3]. However, since the considered grid network for simulation does not have regular spacing as assumed in [3], the time slots cannot be fully utilized.  $\rho^+$  and  $\rho^-$  are computed using (5.3) and (5.4). The values are computed as  $\rho^+ = 0.0713$  s and  $\rho^- = 0.1738$  s. The time slot length is 1 s and the packet duration is computed using (5.5) to be 0.7549 s. The frame length  $T = 4$  s and hence the throughput of the  $\rho$ -Schedule is  $S_\rho = 4.5(1 - 0.0713 - 0.1738) = 3.3970$ .

- *Computing MILFP Schedule:* The solution to MILFP (5.22) results in unslotted transmission schedule presented in Table 5.2. The schedule does not require explicit transmission slots, instead, the transmission on a link starts at a specific time and lasts for a particular duration. The packet duration is computed and is equal to 1.8981 s for all scheduled links. The frame length  $T = 4.0011$  s is computed, the throughput  $S$  is computed to be 4.2695. There is a significant 25.7% increase in the throughput when compared to the throughput computed for  $\rho$ -Schedule.

We perform the simulation after implementing the schedules presented in Table 5.1 and Table 5.2 for a duration of 15 minutes. For example, if the frame length is  $T = 4$  s, with 9 transmissions and receptions per frame, then within a duration of 15 min, the number of frames repeated is  $\frac{15 \times 60}{4} = 225$  frames. Therefore, the number of transmitted packets is computed as  $225 \times 9 = 2025$  packets. The simulator provides the statistics on the number of transmitted and received packets as well as lost packets, in order to compute

the throughput. Given the frame length, packet duration and the number of transmissions per frame the expected throughput can be computed using the number of transmissions as computed above. The expected throughput is then compared with the actual throughput achieved in the simulator. The offered load is increased and the expected throughput and achieved throughput are compared as shown in Fig. 5.8. In the simulation, the offered load is increased by increasing the packet duration of each transmission while maintaining all packet lengths to be equal. As the offered load increases, the throughput increases and reaches a different maximum value for the MILFP schedule and the  $\rho$ -Schedule. The packet duration at which the maximum throughput is achieved matches the analytical throughput computed for both cases. The sharp drop in throughput is due to the sudden increase in the number of packet collisions, attributed to the fact that even a partial overlap in the packets can cause a failure in the reception of both the packets. From Fig. 5.8, we can conclude that the unslotted schedule better utilizes the long propagation delays in the grid network considered.

#### 5.4.2 Robust unslotted schedules

In Section 5.4.1, a grid network is deployed and the deployed locations and propagation delay between nodes are known with certainty. Now we consider the problem with uncertainties in the values of propagation delay between nodes. We assume that the propagation delay  $\hat{D}_{jk}$  on link  $(j, k)$  can be any value in the set  $\mathcal{D}_{jk}$ . As discussed in Section 5.1, the uncertainty in the propagation delays can be characterized by defining the node  $i$ 's location to lie in a Euclidean ball  $B(\mathbf{x}_i, r_i)$ . For given  $r_i \forall i \in \{1, \dots, N\}$ ,  $\mathcal{D}_{jk} \forall (j, k) \in \mathcal{L}$ , the robust MILFP

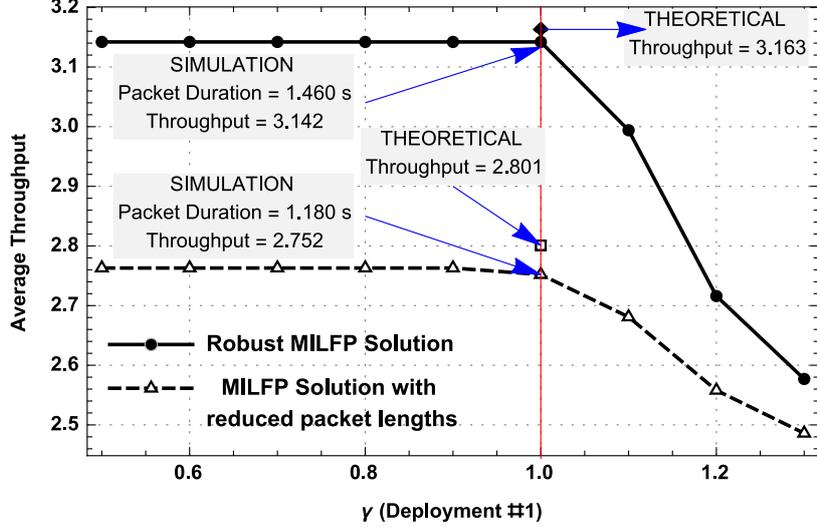
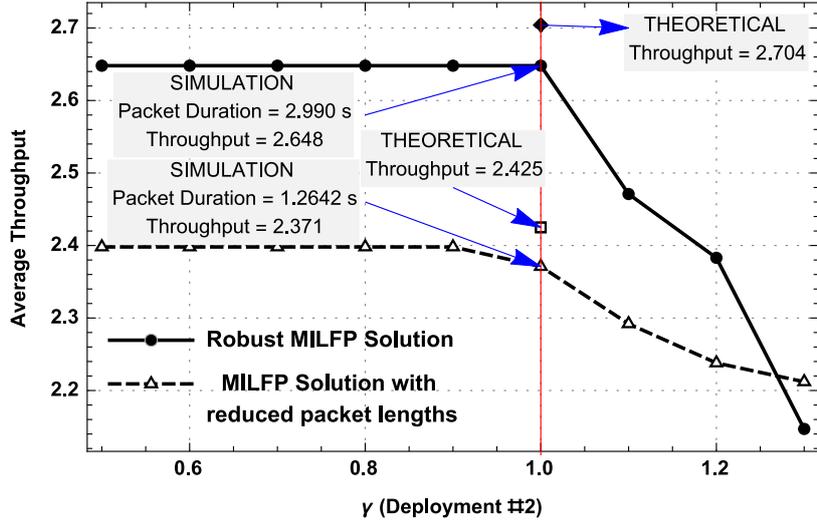
(a) Deployment #1 with  $\frac{r_{\max}}{c} \leq 0.2$ .(b) Deployment #2 with  $\frac{r_{\max}}{c} \leq 0.25$ .

Figure 5.9: Two different grid network deployment scenarios are considered. For each deployment, the uncertainties in the propagation delay are induced. 100 random instances with  $r_{\max} = 0.2$  and  $0.25$  are considered. The average throughput is computed and plotted against the randomness factor  $\gamma$ .

shown in (5.35) can be solved for any particular set of propagation delay values.

To demonstrate the usefulness of the robust unslotted schedules we perform the following steps:

- *Step 1:* We compute the robust unslotted schedule using (5.35) for a

particular set of values of propagation delay lying within the bounds as shown in (5.28). This schedule is verified to be robust to the variations in the values of propagation delay within the uncertainty sets. The corresponding throughput computed must remain same for all such variations in propagation delays.

- *Step 2:* We compute the unslotted schedule using (5.22) for the same set of values of the propagation delay for which the robust schedule was computed in the first step. To make the schedule computed using (5.22) robust to the uncertainties in propagation delays, we reduce the packet duration of the transmissions by  $\frac{2r_{\max}}{c}$ , where  $r_{\max} = \max_{i \in \{1, \dots, N\}} r_i$  and  $c$  is the speed of sound underwater. This schedule is verified to be robust to the uncertainties and the corresponding throughput is compared against the throughput of robust unslotted schedules computed in Step 1.

The values  $r_i \leq r_{\max}$ ,  $\forall i \in \{1, \dots, N\}$  are chosen randomly. We consider a 12-node 3-line multi-hop grid network again and the bounds of the uncertainty sets on each link can be calculated using (5.30) and (5.31).

### Verification in the simulator

We consider two different 12-node 3-line grid network deployments (deployment #1 and #2) with  $\frac{r_{\max}}{c} \leq 0.2$  s and 0.25 s. The values of  $\frac{r_i}{c}$  are chosen randomly to lie between 0 and 0.2 in the first deployment and to lie between 0 and 0.25 in the case of the second deployment. The delay matrix  $\mathbf{D}$  of size  $12 \times 12$  is computed using the deployed locations. For this particular delay matrix, the schedule is computed using (5.35). The schedules are presented in Table 5.3. The packet

TABLE 5.3: ROBUST UNSLOTTED SCHEDULES

Link	Deployment #1				Deployment #2			
	Robust schedules using (5.28)		Robust schedule using (5.23)		Robust schedule using (5.28)		Robust schedules using (5.23)	
	Transmission time (s)	Packet duration (s)	Transmission time (s)	Packet duration (s)	Transmission time (s)	Packet duration (s)	Transmission time (s)	Packet duration (s)
(1, 4)	1.4577	1.4603	3.2690	1.1881	2.700	2.9962	5.3284	1.2642
(2, 5)	3.5411	1.4603	1.3536	1.1881	8.5410	2.9962	3.5834	1.2642
(3, 6)	5.2728	1.4603	3.3420	1.1881	13.1172	2.9962	5.3298	1.2642
(4, 7)	5.0318	1.4603	2.7068	1.1881	0	2.9962	0.0639	1.2642
(5, 8)	2.1626	1.4603	0	1.1881	15.6686	2.9962	1.5311	1.2642
(6, 9)	4.3463	1.4603	2.1651	1.1881	10.6417	2.9962	0	1.2642
(7, 10)	3.9532	1.4603	1.4051	1.1881	7.3122	2.9962	3.8763	1.2642
(8, 11)	5.2736	1.4603	2.6656	1.1881	13.2612	2.9962	5.3434	1.2642
(9, 12)	3.2912	1.4603	1.1949	1.1881	8.3189	2.9962	2.6156	1.2642
Frame length	$T = 4.1551$ s		$T = 3.8164$ s		$T = 9.9696$ s		$T = 4.6901$ s	
Throughput	$S = 3.163$		$S = 2.801$		$S = 2.704$		$S = 2.425$	

duration and the transmission times are listed. Similarly, the schedule is also computed using (5.22). Note that the solution computed using (5.22) is not robust. In order to make the solution robust we reduce the packet duration in the solution by an amount equivalent to the maximum change that can occur in the values of propagation delay, which is given by  $\frac{2r_{\max}}{c}$ . The value of  $\frac{r_{\max}}{c}$  is 0.1579 s for deployment #1 and 0.2254 s for deployment #2. Note that  $r_{\max}$  corresponds to the largest Euclidean ball around the nodes considered in the network. We represent each radius  $r_i$ ,  $\forall i \in \{1, \dots, N\}$  as an element of vector  $\mathbf{r}$ . We introduce a randomness factor  $\gamma$ , which scales the vector  $\mathbf{r}$ , i.e.,  $\gamma\mathbf{r}$ . For example, by varying the value of  $\gamma$  from 0.5 to 1.3, we vary the size of the Euclidean balls. Note that the robust schedule computed are expected to work for values of  $\gamma \leq 1$ , since the solution computed is robust to the variations in propagation delay values corresponding to the elements in  $\mathbf{r}$ . For  $\gamma > 1$ , the corresponding propagation delay among the nodes fall outside the bounds

of the uncertainty sets for which the schedule computed is not robust. The average throughput as a function of  $\gamma$  is plotted in Fig. 5.9. For each value of  $\gamma\mathbf{r}$ , 100 random instances of 12-node 3-line grid networks are generated, the robust schedule computed is verified and the resulting average throughput is computed. For example, in the case of deployment #1, the frame length is  $T = 4.1551$  using (5.35). The theoretical throughput is computed using (5.18) to be  $\frac{1.4603 \times 9}{4.1551} = 3.163$ . Similarly, the throughput for the other cases is computed and listed in Table 5.3. In the simulation, the packets are transmitted and received for a duration of 15 minutes. For example, for the case of deployment #1, the packet duration is set to 1.460 s. Since the frame length is 4.1551 s, there will be  $\frac{15 \times 60}{4.1551} = 216.6$  frames for the duration of simulation. Since, there are 9 successful transmissions per frame we expect to see  $216.6 \times 9 = 1949$  successful transmissions and receptions. We note from the logs of the simulation that there are 1937 successful receptions with zero collisions. The difference in the transmission and reception packets comes from the initial period in which the receptions have not yet started. Therefore, the throughput in the simulation is computed as  $\frac{1937 \times 1.460}{15 \times 60} = 3.142$ . As shown in Fig. 5.9, the robust schedules clearly perform better than the schedules that are computed using (5.22) with reduced packet duration.

## 5.5 Summary

We considered unslotted and robust transmission schedules that utilize the large propagation delays in practical multi-line grid networks with multi-hop relaying to provide throughput-maximizing schedules in the presence of propagation

delay uncertainties. We demonstrated that the unslotted schedules computed by solving MILFP always provide schedules which perform better than the  $\rho$ -Schedules proposed recently. The propagation delays are better utilized when the packet transmissions are not restricted to the time slots. The inefficiencies caused due to approximations in non-integer propagation delays and the resulting guard times are mitigated by using unslotted schedules. Moreover, the uncertainties in the propagation delays are considered in the model and a robust MILFP is formulated. We verified the robust schedules in the simulator. This study provides two substantial results with regards to multi-line grid topologies with multi-hop relaying. The first is the technique to compute unslotted schedules which better utilizes the propagation delay information and provide throughput-maximizing schedules which are closer to the upper bound. The second includes finding robust schedules in the presence of propagation delay uncertainties. These results provide solutions which can be used for various applications involving the multi-hop grid networks. Note that all the simulations result in the same conclusions even with the 3D deployments.

## Chapter 6

### Modem Constraints and Experimental Demonstration

---

In the previous chapters, we have presented transmission strategies which can better exploit large propagation delays in UWA networks. In this chapter, we consider the practical challenges in implementing such protocols on underwater acoustic modem. We present the design, implementation and testing of *Super-TDMA*<sup>1</sup> protocol which exploits the large propagation delay in underwater acoustic channels by aligning most of the interference by the unintended messages in time domain. The work presented in this chapter is published in [105], [111]. Specifically, we report the following contributions in this chapter:

- First, we study the practical modem constraints on underwater acoustic modem. The inclusion of modem constraints results in the design which can be realized in practice. The optimal packet and time slot lengths are computed which result in maximum utilization of the time slots by minimizing the guard times.
- Next, we present the implementation challenges of Super-TDMA on an underwater acoustic modem and the experimental results demonstrating interference alignment, the crossing of simultaneously transmitted packets

---

<sup>1</sup>*Super-TDMA* was first termed in [110] to represent a concept - a form of TDMA that can utilize propagation delay unlike traditionally done.

in water, and the time synchronization among the deployed nodes in the network.

In [112], two practical issues caused by the underwater acoustic modems, low available transmission rates and long preambles are considered. However, there are other problems which are important when timing-sensitive protocols are to be implemented on the modem. The schedules exploiting large propagation delays demand the frequent transitions between transmission and reception times. This results in the frequent switching between transmission and reception modes in the modem. The time slots and the guard intervals need to be chosen carefully to maximize the utilization efficiency of the time slot, and hence the switching times are important in selecting the time slot lengths and in minimizing the guard times. The switching times may vary for different modems [99],[113],[114]. We present the optimization problem for selecting the time slot lengths to be used considering the practical modem constraints. The packet lengths are set to be the integer multiple of smallest incremental packet duration that can be set in the modem, which will effect the time slot length chosen. However, this assumption can be relaxed by considering the set of allowable packet lengths in the modem.

### **6.1 Transition Times in Modem**

The transitions between transmission and reception modes in the modem result in processing delays. The minimum time that is required for setting up the modem to transmit or receive correctly during different transitions between these modes are illustrated here.

1. **RX-TX transition:** The processing delay to switch from reception (RX) mode to transmission (TX) mode in modem is denoted by  $t_{\text{RX-TX}}$ . For example, to set up a packet transmission in UNET-II modem [99] from the default state which is receiver enabled (RX) state, the receiver needs to be disabled followed by switching ON the power amplifier for transmission, which contributes to  $t_{\text{RX-TX}}$ .
2. **TX-TX transition:** To set up a packet transmission immediately after the transmission in the previous slot, the power amplifier state need not be changed and can be left ON. This transition time is denoted by  $t_{\text{TX-TX}}$ .
3. **TX-RX transition:** To prepare the modem for reception after the packet transmission in the previous slot, power amplifier needs to be switched OFF and the receiver is enabled. The processing delay in modem to switch from RX mode to TX mode is denoted by  $t_{\text{TX-RX}}$ .
4. **RX-RX transition:** A packet after the reception in the previous slot, can be set up after  $t_{\text{RX-RX}}$  amount of time.

The values of these parameters may vary for different modems which will constitute different transition times. Since these times can be measured and are known a priori, we can fix these parameters in the optimization problem presented next. The start and the end guard times need to be atleast greater than the time that it takes to prepare the modem for either transmitting or receiving the packet correctly. The guard time at the start of the time slot denoted by  $t_s$  need to be greater than the largest among the transition times  $t_{\text{TX-TX}}$  and  $t_{\text{RX-TX}}$ , which are the times before a transmission mode can be set

in the modem from different states, i.e.,

$$t_s > \max\{t_{\text{RX-TX}}, t_{\text{TX-TX}}\}. \quad (6.1)$$

Similarly, after a packet transmission in the previous slot, the largest time to be waited for to set up the reception in modem is given by  $\max\{t_{\text{TX-RX}}, t_{\text{RX-RX}}\}$  and hence it is enough to set

$$t_e > \max\{t_{\text{TX-RX}}, t_{\text{RX-RX}}\}. \quad (6.2)$$

Note that (6.1) and (6.2) do not take into consideration the propagation delays. These constraints consider only the modem related constraints and enable the solution to impose guard times which then result in implementable schedules on the modem.

## 6.2 Optimization Problem

### 6.2.1 Without modem constraints

Consider a randomly deployed  $N$  node UWA network represented by its corresponding non-integer delay matrix  $\mathbf{D}$ . The approximate integer delay matrix needs to be computed by selecting time slot length  $\tau$  as a result of which the values of  $\rho^+$  and  $\rho^-$  are set. The  $\rho$ -throughput  $S_\rho$  is given by the number of successful receptions per time slot multiplied by the slot utilization efficiency as shown in (2.10). Now, we compute the slot utilization by considering only the payload duration. Therefore, the  $\rho$ -throughput  $S_\rho$  considering only the payload

duration is given by

$$S_\rho = S(1 - \rho^- - \rho^+ - \frac{t_p}{\tau}) \quad (6.3)$$

where  $t_p$  is the packet header duration. The optimization problem in this case is formally written as:

$$\begin{aligned} & \underset{\tau}{\text{maximize}} && S(1 - \rho^- - \rho^+ - \frac{t_p}{\tau}) \\ & \text{subject to} && \tau = \{\tau_{\min} + m\Delta x ; m \in \{1, \dots, \lfloor \frac{\tau_{\max} - \tau_{\min}}{\Delta x} \rfloor\}\}. \end{aligned} \quad (6.4)$$

For each value of the time slot length  $\tau$ , we compute the values of  $\rho^-$  and  $\rho^+$  and compute  $S$  from the schedule  $\mathbf{W}^{(T)}$  computed using the scheduling algorithm from [1]. The time slot length corresponding to the maximum value of the objective function is chosen to be the optimal time slot length  $\tau^*$  and corresponding packet duration is set.

### 6.2.2 With modem constraints

The optimization problem to compute the optimal value of the time slot length is now modified to include the modem constraints presented.

$$\begin{aligned} & \underset{\tau}{\text{maximize}} && S(1 - \rho^- - \rho^+ - \frac{t_p}{\tau}) \\ & \text{subject to} && \tau\rho^- > \max\{t_{\text{RX-TX}}, t_{\text{TX-TX}}\} \\ & && \tau\rho^+ > \max\{t_{\text{TX-RX}}, t_{\text{RX-RX}}\} \\ & && \tau = \{\tau_{\min} + m\Delta x; m \in \{1, \dots, \lfloor \frac{\tau_{\max} - \tau_{\min}}{\Delta x} \rfloor\}\}. \end{aligned} \quad (6.5)$$

The solution of this problem provides the best utilization efficiency considering the modem constraints. Note that the constraints need to be generic enough so

that the parameters can be measured for other practical modems used today. For example, we considered four main parameters; TX-RX, TX-TX, RX-TX and RX-RX switching times. For any practical modem, these parameters can be measured and might vary based on the hardware and software architecture and implementation. Moreover, the preamble duration is another parameter which also plays a part in the objective of the optimization problem and is different for different practical modems. Once, these parameters are measured and the optimization problem is set up, the solution to the problem is implementable on the modem for which the parameters were used.

### 6.3 Case Study

The UNET network deployed (see Fig. 4.2) during the MISSION 2013 experiment in Singapore waters consisted of a UNET-II modem [99] (node 21) mounted below a barge and six UNET-PANDA nodes [100] (nodes 22,27, 28, 29, 31 and 34) deployed at various locations within a  $2 \times 2$  km area around the barge.

We consider a network geometry from this experiment as shown in Fig. 4.2, with distance matrix:

$$\mathbf{L} = \begin{bmatrix} 0 & 599 & 932 \\ 599 & 0 & 944 \\ 932 & 944 & 0 \end{bmatrix}$$

and apply the technique to find the optimal time slot lengths, visualize the schedule computed and verify the performance in the simulator with and without modem constraints. The modems labeled as P21, P28 and P29 in Fig. 4.2 are considered as node 1, node 2 and node 3 respectively in the analysis.

TABLE 6.1: SOLUTION WITHOUT MODEM CONSTRAINTS

Parameter	Value without modem constraints
$\tau^*$	204 ms
$t_{pd}$	184 ms
$t_s$	19.03 ms
$t_e$	0.97 ms
$S_\rho$	1.205

### 6.3.1 Results

The optimization problem is solved to compute the optimal time slot length with and without modem constraints for the considered network geometry.

#### Without modem constraints

The time slot length  $\tau$  is varied from  $\tau_{\min} = 1$  ms to  $\tau_{\max} = 3000$  ms,  $t_p = 20$  ms and  $\Delta x = 1$  ms is set. The optimal value of the time slot length along with other parameters are presented in Table 6.1. The delay matrix and the integer delay matrix corresponding to time slot length  $\tau^*$  are

$$\mathbf{D} = \frac{\mathbf{L}}{c\tau^*} = \begin{bmatrix} 0 & 1.9067 & 2.9666 \\ 1.9067 & 0 & 3.0048 \\ 2.9666 & 3.0048 & 0 \end{bmatrix}, \quad \mathbf{D}' = \begin{bmatrix} 0 & 2 & 3 \\ 2 & 0 & 3 \\ 3 & 3 & 0 \end{bmatrix}.$$

For this delay matrix, the optimal schedule is computed using the algorithm presented in [1]. The schedule is

$$\mathbf{W}^{(8)} = \begin{bmatrix} 3 & 2 & -3 & 3 & -2 & -3 & -2 & 2 \\ -3 & -1 & 1 & -1 & 1 & 3 & 3 & -3 \\ -2 & -2 & 1 & -1 & 2 & 2 & -1 & 1 \end{bmatrix}.$$

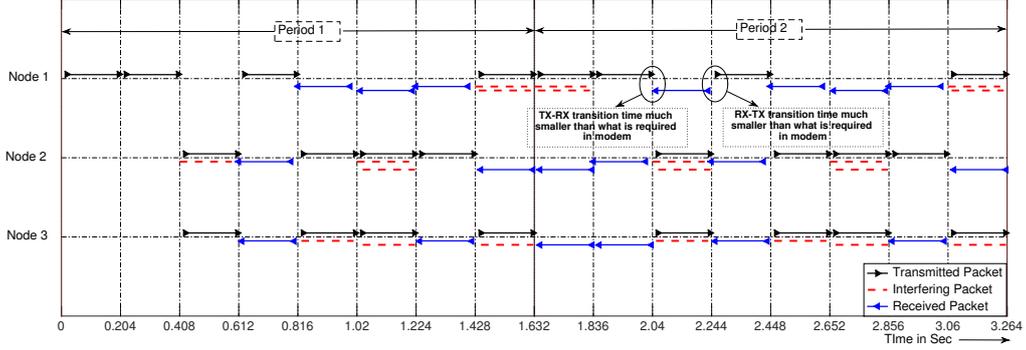


Figure 6.1: Schedule visualization without modem constraints (the guard times set in the time slot take care of only the early receptions and delayed receptions problem due to the approximations made in the delay matrix because of the network geometry).

This schedule is visualized in Fig. 6.1, using the optimal values computed for the network setting. Since we know the time slot length, guard times and period of the schedule, i.e., 8 slots, we can plot the transmitted packets, received packets and the interfered packets accurately. We leave  $t_s$  amount of time before the transmission in a transmitting slot and  $t_e$  amount of time at the end. We observe that all the receptions in the schedule are interference free as expected. Note that in Fig. 6.1, the gap between end of the packet transmission in previous slot and start of the reception in next slot is very small at several time instances and in practice while implementing on modem this may not be achievable and hence the modem constraints need to be added to the optimization problem.

**Effect of modem constraints on schedule computed:** In order to implement the schedule  $\mathbf{W}^{(8)}$  on the modem, the packet lengths need to be reduced and enough guard times need to be left in order to configure the modem correctly for transmission and reception. We set  $t_{\text{TX-RX}} = 20$  ms,  $t_{\text{RX-TX}} = 70$  ms,  $t_{\text{TX-TX}} = t_{\text{RX-RX}} = 0$  ms. For this setting,  $t_s > 70$  ms

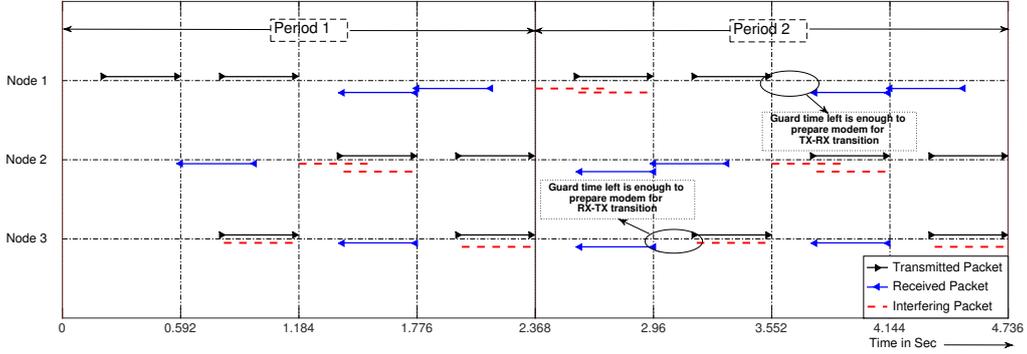


Figure 6.2: Schedule visualization with modem constraints (the guard times set in the time slot not only take care of the constraints due to geometry but also consider TX-RX and RX-TX transition times in modem).

TABLE 6.2: FEASIBLE SOLUTION WITH MODEM CONSTRAINTS

Parameter	Value with modem constraints
$\tau^*$	204 ms
$t_{pd}$	114 ms
$t_s$	70 ms
$t_e$	20 ms
$S_\rho$	0.6911

and  $t_e > 20$  ms can be computed from equations (6.1) and (6.2). Hence, the maximum packet length that can be used without collisions with this schedule satisfying modem constraints is  $t_{pd} = \tau^* - t_s - t_e = 114$  ms. Therefore,  $t_{payload} = 94$  ms and the schedule computed  $\mathbf{W}^{(8)}$  consists of 12 receptions in 8 time slots. Hence,  $S = \frac{12}{8} = 1.5$ . We can now compute the  $\rho$ -throughput,

$$S_\rho = S\eta = 1.5\left(\frac{94}{204}\right) = 0.6911.$$

The different parameter values computed considering the modem constraints to implement the schedule  $\mathbf{W}^{(8)}$  on modem are presented in Table 6.2. Note that although in theory the throughput computed for this schedule is 1.204, in order

TABLE 6.3: OPTIMAL SOLUTION WITH MODEM CONSTRAINTS

Parameter	Value with modem constraints
$\tau^*$	592 ms
$t_{pd}$	368 ms
$t_s$	203.05 ms
$t_e$	21.01 ms
$S_\rho$	0.8817

to implement on the modem, the packet length is reduced and guard intervals are left and hence the effective achievable throughput is 0.6911.

### With modem constraints

If the modem constraints are considered in the optimization problem, we can find an optimal schedule which when implemented results in higher throughput than what is computed in the previous section. The time slot  $\tau$  in this case is varied from  $\tau_{\min} = 1$  ms to  $\tau_{\max} = 3000$  ms,  $t_p = 20$  ms and  $\Delta x = 1$  ms is set as in previous section, also  $t_{RX-TX} = 70$  ms,  $t_{TX-RX} = 20$  ms and  $t_{TX-TX} = t_{RX-RX} = 0$  ms are set. The optimal time slot length to be used are computed satisfying all the modem constraints. The optimal values are listed in the Table 6.3. The delay matrix and integer delay matrix for this optimal time slot length are given by:

$$\mathbf{D} = \begin{bmatrix} 0 & 0.6570 & 1.0223 \\ 0.6570 & 0 & 1.0355 \\ 1.0223 & 1.0355 & 0 \end{bmatrix}, \quad \mathbf{D}' = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}.$$

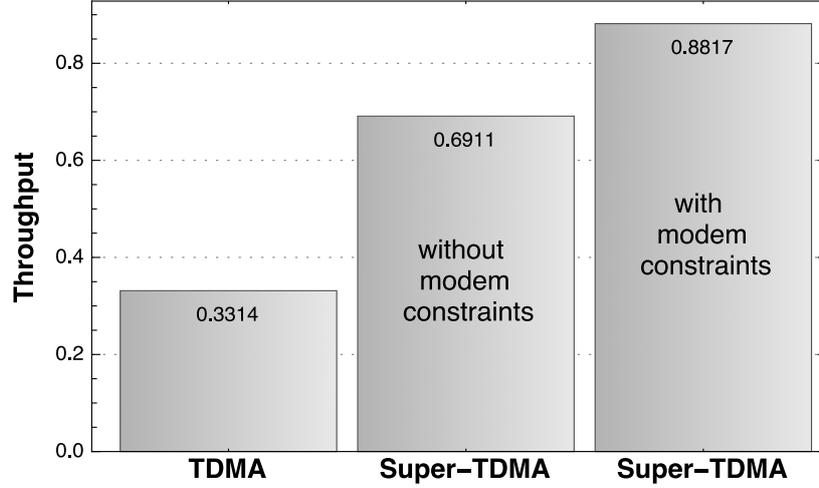


Figure 6.3: Throughput comparison of Traditional-TDMA and Super-TDMA protocols with and without modem constraints for the considered network geometry.

For this delay matrix  $\mathbf{D}'$ , the optimal schedule is computed:

$$\mathbf{W}^{(4)} = \begin{bmatrix} 2 & 3 & -3 & -2 \\ -3 & -1 & 1 & 3 \\ -2 & 1 & -1 & 2 \end{bmatrix}.$$

Note that for the same network geometry, due to the modem constraints a different schedule is adopted in this case with a period  $T = 4$ .

The scheduled transmissions and the receptions are visualized in Fig. 6.2. Enough time is available between transmission and reception of the packets to configure the modem. Note that  $t_s = 203.05 \text{ ms} > t_{\text{RX-TX}}$ . Also,  $t_e = 21.01 \text{ ms} > t_{\text{TX-RX}}$ .

**Traditional-TDMA and Super-TDMA - Comparison:** The throughput

of *Traditional*-TDMA is given by:

$$S_{\text{TDMA}} = \frac{t_{\text{payload}}}{t_{\text{pd}} + t_{\text{maxpd}}}$$

where,  $t_{\text{maxpd}}$  is the maximum propagation delay among the links considered in the network and  $t_{\text{pd}}$  is the packet duration given by (4.22). Also, the time slot length  $\tau = t_{\text{pd}} + t_{\text{maxpd}}$ . If TDMA is implemented on the modem, enough guard times need to be included in the time slot, therefore,

$$S_{\text{TDMA}}^{\text{modem}} = \frac{t_{\text{payload}}}{t_{\text{pd}} + t_{\text{maxpd}} + t_{\text{RX-TX}}} = 0.3314.$$

The throughput for Traditional-TDMA and Super-TDMA with and without considering the modem constraints are presented in Fig. 6.3. The poor performance of TDMA is because of the presence of large propagation delays in underwater acoustic networks. However, Super-TDMA protocol exploits the large propagation delay in order to improve the throughput.

## 6.4 Implementation & Experimental Demonstration

We deployed a 3 node UWA network in Singapore waters and solve the problem (6.5) for the resulting network geometry to compute the schedule along with the time slot length and the guard periods and implement it on all three nodes along with time synchronization. In Section 6.4.1, we present the practical Super-TDMA schedule computed for the 3 node network deployed. Next, we present a detailed overview of the implementation and challenges in Section 6.4.2, followed by the experimental results in Section 6.4.3.

TABLE 6.4: TRANSMISSION SCHEDULE FOR 3 NODE LINEAR NETWORK

	Time Slot 1	Time Slot 2	Time Slot 3
<b>Node 1</b>	TX to 3	TX to 2	RX from 3
<b>Node 2</b>	TX to 3	IDLE	RX from 1
<b>Node 3</b>	TX to 1	RX from 2	RX from 1

TABLE 6.5: DISTANCE MATRIX OF THE SEA-TRIAL 3 NODE NETWORK

	Node 31	Node 21	Node 22
<b>Node 31</b>	0	807 m	1574 m
<b>Node 21</b>	807 m	0	783 m
<b>Node 22</b>	1574 m	783 m	0

#### 6.4.1 Practical Super-TDMA schedule

We deployed a network (see Fig. 6.4) during a recent at-sea experiment in Singapore waters consisting of a UNET-II modem [99] (node 21) mounted below a barge and two UNET-PANDA nodes [100] (nodes 22 and 31) deployed at locations as shown in Fig. 6.4. We considered node 31, node 21 and node 22 forming a 3 node network for demonstrating Super-TDMA and denote them as node 1, node 2 and node 3 respectively in the analysis. Distances among the nodes measured are represented in Table 6.5 as a distance matrix, where each cell of the table represents the distance between the corresponding nodes in its row and column. Considering the speed of sound,  $c = 1540$  m/s, propagation delays are computed and represented in the form of a delay matrix in units of optimal time slot length computed by solving problem (6.5):

$$\mathbf{D} = \begin{bmatrix} 0 & 1.0207 & 1.9908 \\ 1.0207 & 0 & 0.9903 \\ 1.9908 & 0.9903 & 0 \end{bmatrix} \approx \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix}. \quad (6.6)$$



Figure 6.4: UNET network node locations during the experiment in Singapore waters. Yellow markers are network nodes. The links considered for demonstrating Super-TDMA are marked with distances between them.

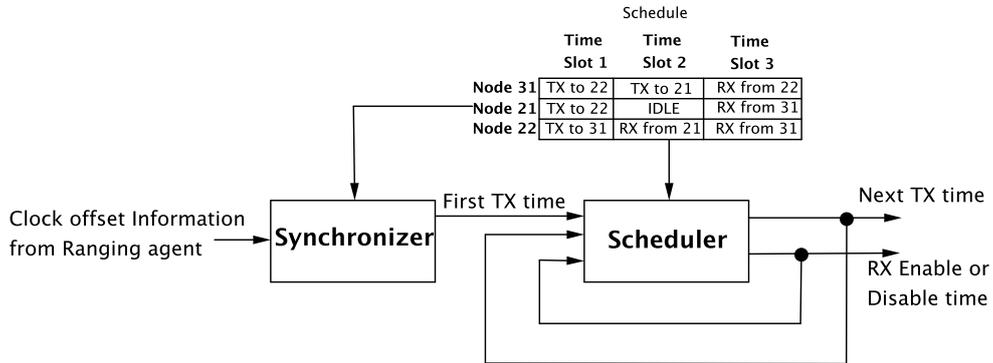


Figure 6.5: Overview of the implementation showing the modules: synchronizer and scheduler. Both the modules are implemented on all 3 nodes with the objective to first achieve the time synchronization among the nodes and second to prepare the modems for accurately transmit and receive at scheduled times.

The optimal time slot length is computed to be 514 ms. Note the approximated integer delay matrix in (6.6) represents a linear network and hence we adopt the transmission schedule as shown in Table 6.4 and implement it on all three nodes. The packet duration is computed as 498 ms, and the guard times to be

CHAPTER 6. MODEM CONSTRAINTS AND EXPERIMENTAL DEMONSTRATION

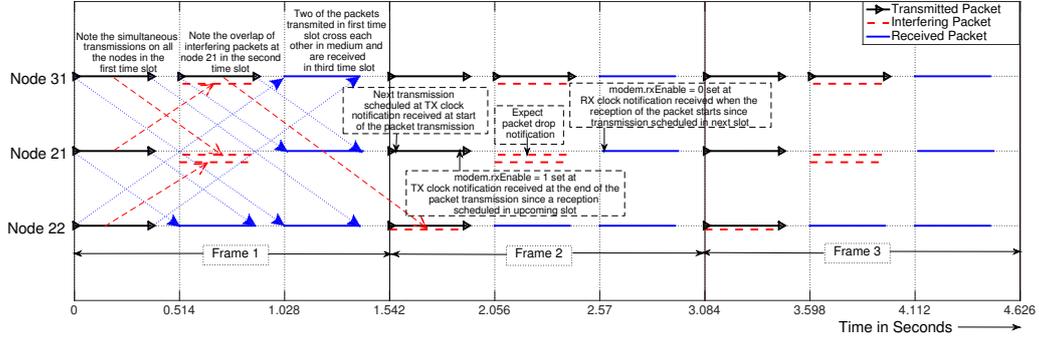


Figure 6.6: Visualization of the expected scheduled events in time is shown. Note the crossing of the transmitted packets in the medium, interference alignment in the transmitting slots, zero interference during receiving slots and the scheduler job sequence to configure the modems to switch between transmission and reception modes at appropriate times.

left at the start and end of the transmissions are computed to be 10.63 ms and 4.98 ms respectively (see [105] for details). Note that such small guard times are not sufficient for modem switching times; these guard times only mitigate the effect of approximations made in propagation delays. For the purpose of this demonstration, we implement the schedule on UNET-II modems, for which the minimum guard period between two successive accurately timed-transmissions was measured to be approximately 120 ms. This duration is sufficient to configure the modem correctly for transmissions and receptions. Therefore, in the implementation, we use the time slot length to be 514 ms and select a packet duration which is close to  $498 - 120 = 378$  ms. Hence, we expect a throughput,  $S_\rho = \frac{4}{3} \times \frac{378}{514} = 0.97$ . The transmission schedule is shown in Table 6.4 with the time slot length 514 ms and the packet duration set to 368 ms (closest packet duration setting available) is visualized in Fig. 6.6. The expected transmission and reception events are plotted in time. Note that all the time slots with packet

receptions are interference-free and the interfering packets are aligned with the transmitting slots. After collecting the data from the experiment at sea, we plot the transmission and reception events as they happen in time on all the nodes and visualize it to compare it with the expected result as shown in Fig. 6.7.

#### 6.4.2 Overview of the implementation

The implementation of Super-TDMA schedules on the modem consists of two modules: synchronizer and the scheduler as shown in Fig. 6.5. The objective of the synchronizer is to achieve time synchronization among the nodes whereas the scheduler's job is to schedule various events required by the nodes for accurate transmissions and receptions in the appropriate time slots. The schedule used is the input to both modules as shown in Fig. 6.5.

##### **Time synchronization**

The UNET-II modems are equipped with an oven-controlled crystal oscillator (OCXO) to maintain clock with a resolution of  $1 \mu s$  and also provides a low drift ( $\pm 0.05$  ppm drift) [99] timing reference for ranging, MAC and network protocols that need accurate time synchronization. We implement the synchronizer module to take into account the clock offset measured among the nodes. We do not compensate for clock skew since the drift is negligible over the period of this demonstration. The **Ranging** agent in UnetStack [109] uses two-way travel time to estimate range and clock offset between a pair of nodes. We use the **Ranging** functionality and estimate the clock offset between every pair of nodes. The most advanced node in time then broadcasts a time in future (computed on its local clock) to all other nodes and is used to schedule their first transmission.

**CHAPTER 6. MODEM CONSTRAINTS AND EXPERIMENTAL DEMONSTRATION**

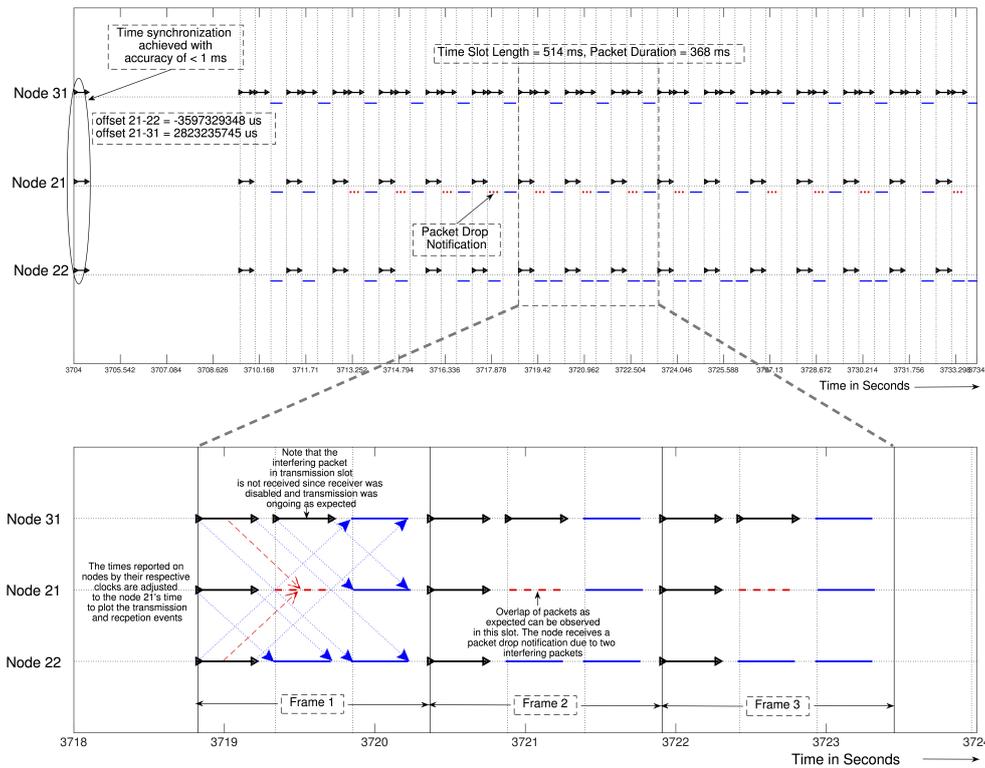


Figure 6.7: Visualization of the actual events as happened at sea in time is shown. Note the crossing of the transmitted packets in the medium. The intended overlap of the interfering packets happened as expected in the corresponding time slots. The time synchronization is achieved perfectly as can be observed by looking at the first packets that are transmitted on all three nodes.

Synchronization accuracy achieved during the demonstration was 181  $\mu$ s.

**Scheduler**

The scheduler configures the nodes at accurate times for transmissions and receptions in appropriate time slots. Since the time slot length and the packet duration are in the order of the propagation delay among the nodes, frequent switching from transmission to reception modes and vice versa is required and is challenging while implementing these schedules on the modem. For the schedule considered for this experiment, the time slot length is 514 ms. This demands

the two successive transmissions in the first two time slots on node 31 to be 514 ms apart accurately. To achieve this objective, we use timed transmissions (see [108]) in UNET-II modem which allows us to schedule transmissions ahead of time and ensure the desired accuracy in transmission times.

UnetStack provides *notification* messages on successful transmissions and receptions. These can be used as event triggers to configure the modem correctly at appropriate times. Specifically, the notification messages received by the SuperTDMA agent at the start of the transmission, end of the successful transmission and start of the packet reception are utilized. A packet transmission is scheduled ahead of time at the start of the preceding packet transmission when the corresponding notification is received (see Fig. 6.6). Note that waiting until the end of the transmission reduces the time slot utilization and hence a feature allowing transmissions to be scheduled at the start of the packet transmission is useful. Similarly, when the modem is expected to receive a packet in the next time slot, the receiver is enabled at the notification message received at the end of the preceding packet transmission. In Fig. 6.6, we show this in the first time slot of the second frame, where node 21 transmits a packet to node 22 in the first time slot. At the notification received indicating the start of the packet transmission, the next transmission is scheduled, which for node 21 is 3 slots ahead in time. Moreover, at the end of the packet transmission, the receiver is enabled since a reception is scheduled in the upcoming slot. Note that for this particular case, the second time slot expects two interfering packets and since the receiver is enabled, the intended overlap of the interfering packets is expected. A packet drop notification message is received, as expected when the

two packets overlap. Note that, it is necessary to disable the receiver during the transmitting slot since the schedules are designed such that the interfering packets overlap with the transmitting slots. In case, the receiver is not disabled, it is possible that modem starts receiving the unintended interference message before it starts to transmit.

### 6.4.3 Experimental results

The schedule for a linear network along with the time slot length of 514 ms and the packet duration of 368 ms was used in the experiment. UnetStack allows logging of the notifications (carrying transmission and reception times) sent by the appropriate agents in the event of transmissions and receptions. These times are used to plot the transmission and reception events to visualize the schedule in Fig. 6.7. The zoomed-in events are also plotted to compare the result with the expected schedule plotted in Fig. 6.6. The packets cross in water and are received successfully at the intended nodes while the interference is aligned with the transmitting slots and the nodes are correctly configured to transmit in those slots. Note the time slot 2 at node 21, is expected to be an IDLE slot and receives two interfering packets at the same time. It is expected to receive packet drop notification message at this time slot. We mark this in Fig. 6.7, the packet drop notification is indeed received in this time slot. For the demonstrated schedule, the throughput is computed as  $S_p = \frac{4}{3} \times \frac{368}{514} = 0.95$ . Super-TDMA requires frequent switching of transmission and reception modes in the modem.

## 6.5 Summary

The switching times in modem between transmission and reception modes play a critical role in selecting the optimal time slot lengths. The optimal setting of time slot length minimize the guard times for a schedule and eventually maximize the slot utilization efficiency. It is shown that if not taken into consideration, it can severely degrade the performance of Super-TDMA. The optimization problem is presented with and without modem constraints and its impact on the schedule is shown by visualizing the schedule in both the cases. Lastly, we also presented two implementation related problems on modem – time synchronization among nodes in the network and the need for accurate timed transmissions. The implementation is verified on underwater acoustic modems deployed in the Singapore waters for timed transmissions, accurate receptions and time synchronization. The schedule exploiting the large propagation delays, which is computed for the network geometry deployed is used to showcase the impact on the throughput if the design and implementation insights presented can be utilized in such experiments. Moreover, if the underwater acoustic modems are equipped with hardware and software capabilities to achieve better switching times allowing lesser guard periods, the concept of Super-TDMA can prove to be very useful for consideration in future MAC protocols exploiting large propagation delays in UWA networks. This work helps in reducing the gap between theoretical and practical aspects of the protocol presented and brings it one step closer to reality.

## Chapter 7

### Discussion, Conclusions & Future Research

---

#### 7.1 Discussion & Conclusions

The non-negligible propagation delays in UWA networks due to the slower speed of sound in water has remained a challenge for designing efficient MAC protocols. Recent research efforts exploiting or utilizing the propagation delay information in UWA networks showed that significant gains in throughput can be achieved, compared to those from terrestrial wireless networks. Although the throughput upper bound for an  $N$  node network is established, the arbitrary networks deployed often cannot achieve this upper bound. We presented in this thesis, strategies to improve the network throughput in arbitrary networks by better utilization of propagation delay information leading to a network throughput much closer to the upper bound  $\frac{N}{2}$ .

The first strategy to improve throughput in arbitrary network geometries adopted time-slotted schedules. By controlling the transmission power we limited the interference range, thereby increasing the network throughput significantly. This allows more transmission opportunities in the network resulting in throughput that is closer to the upper bound  $\frac{N}{2}$  when compared with the throughput of that without power control. We also showed instances of random network geometries which are able to achieve the throughput upper

bound by means of power control, which is not possible otherwise. These examples presented demonstrate that the transmission power control along with the exploitation of large propagation delays can result in schedules which yield higher throughput.

The second strategy considered the problem of computing unslotted schedules with variable packet lengths. These schedules do not explicitly require the nodes to transmit in particular time slots. Moreover, the set of links to be scheduled along with the packet traffic demands are input to the algorithm, allowing the user-specified links to be scheduled. An optimization problem with the goal of finding throughput-maximizing schedules is presented. The fractional idle time in a frame is minimized. In contrast to the approach where the minimum length schedules are computed by minimizing the frame length, we considered the fractional idle time to allow variability in the packet transmission duration. We visualized the computed schedules and presented the comparison between time-slotted, fixed packet duration solution and the proposed MILFP solution with no time-slotting and variable packet duration. The variability in packet duration was crucial in achieving the throughput gain, resulting in throughput closer to the upper bound  $\frac{N}{2}$ . The proposed algorithm outperformed the existing state-of-the-art methods to find schedules exploiting the large propagation delays.

The centralized algorithm proposed resulting in unslotted schedules with variable packet lengths is scalable to much larger practical multi-hop multi-line grid networks. However, for such networks link fairness is a crucial requirement. We therefore proposed a strategy which result in unslotted schedules with

equal packet duration for satisfying link fairness requirement. We modified the objective function to let the packet duration remain equal while maximizing the throughput. We demonstrated that the unslotted schedules computed by solving the proposed MILFP always provide schedules which perform better than the time-slotted  $\rho$ -Schedules. The inefficiencies caused due to approximations in the non-integer propagation delays and the resulting guard times, are mitigated by using unslotted schedules. Moreover, the uncertainties in the propagation delays are considered in the model and a robust MILFP is formulated. We verified the robust schedules in the simulator and demonstrated the robustness to variations in the propagation delay. As a benchmark for comparison, we considered the non-robust schedule computed with reduced packet duration to allow collision-free transmissions in the worst case scenarios. We show that when the nature of uncertainties in the propagation delay of links is more complex, the gain in throughput due to the robust schedules is higher.

The proposed throughput-maximizing strategies perform better due to the concurrent transmission of multiple packets leading to the packets "crossing in flight" and the maximum overlap of interfering packets with the transmission duration. The practical challenges in demonstrating these key concepts through experiments at sea are studied. First, we model the practical constraints in the underwater acoustic modem and compute schedules which are implementable. Next we implement these schedules on a particular underwater acoustic modem to test at sea. The switching times in the modem between transmission and reception modes play a critical role in selecting the optimal time slot lengths. The optimization problem is presented with and without modem constraints

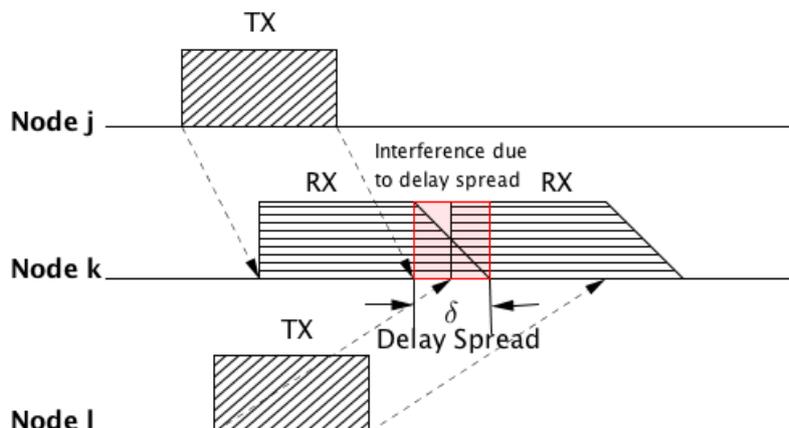


Figure 7.1: Effect of spreading delay on Super-TDMA schedules.

and its impact on the schedule is shown by visualizing the schedule in both the cases. The schedule computed for a 3 node network deployed at sea is used to demonstrate the key concepts. The insights gained from the study of practical challenges can be utilized in design and implementation of such MAC protocols. This work helps to reduce the gap between theoretical and practical aspects of the protocol presented and brings it one step closer to reality.

Note that in this research we have considered TDMA based protocols and a question may arise on how does the proposed Super-TDMA based strategies perform under heavy multipath where the spreading delay may be large. Therefore, we include a brief discussion on the effects of spreading delay on Super-TDMA schedules.

### **Effect of spreading delay on Super-TDMA schedules**

Under heavy multipath, the delay spread of the channel may be large and this might effect the Super-TDMA schedules proposed in this work. For the sake of discussion, assume delay spread of the channel to be  $\delta$  as shown in Fig. 7.1.

Transmission of packet as per the transmission schedule at time  $t_{jk}$  from node  $j$  to node  $k$  will cause the packet to be fully received at the receiver at time  $t + \tau_{jk} + D_{jk} + \delta$  instead of  $t + \tau_{jk} + D_{jk}$ , where  $D_{jk}$  is the propagation delay between node  $j$  and node  $k$  and  $\tau_{jk}$  is the transmission/packet duration. This will cause any other packet which is received after  $t + \tau_{jk} + D_{jk}$  but before  $t + \tau_{jk} + D_{jk} + \delta$  to experience interference due to multipath and might get lost (see Fig. 7.1). However, the propagation delay constraints ((4.1) & (4.2)) can be easily modified to include the known maximum delay spread. This will cause the transmission schedule to cater for enough guard times between the two consecutive receptions. However an important question is whether in practice, the packet lengths employed are comparable to the delay spread of the channel and whether or not the guard times already computed due to the network geometry and modem constraints are enough to deal with the delay spread of the channel ? These questions are answered in the following two points:

1. The delay spread due to the multipath in the channel must be smaller than the guard times between successive packet receptions. The data collected during the various experiments in Singapore waters show that the delay spread in the channel is approximately 10 ms [100]. In practical schedules implemented on the modem with the preamble, the packet duration tend to be much longer of the order few hundreds of milliseconds.
2. The modem constraints included in Chapter 6 resulted in implementable schedules with guard times which were larger than 10 ms for the UNET-II modem. Therefore, in practice, these guard times are already enough to

handle the effect of spreading delay. However, in theory, the delay spread can be included in the problem formulation easily as described earlier.

### **Limitation**

Throughout this work, we have assumed that perfect communication exists between the nodes in the network and the central controller. Since, we have proposed a centralized algorithm, this assumption does not affect the importance of the results presented in this thesis. In the experiments carried out at sea, the schedules were transmitted from the central controller acoustically to other nodes underwater as part of the network initialization. After the network initialization, the nodes start transmitting as per the schedule computed. However, joint topology discovery, network initialization, time synchronization along with optimum transmission schedules merits further investigation.

## **7.2 Future Research**

### **Mobility**

*Limited mobility:* The proposed strategies in Chapter 3 and Chapter 4 considered static nodes with the assumption of knowing the propagation delays with absolute certainty. However, to consider limited mobility of the nodes due to ocean waves/currents or change in sound speed or uncertainty in the measurement of propagation delays, we considered robust unslotted schedules in Chapter 5. We have shown in Chapter 5 that we can indeed find schedules which are robust to the changes in the values of propagation delays if they are known to lie in a certain range.

*Group mobility model:* Another interesting set up is when the nodes in the

network are mobile but move following a group mobility model (i.e., low relative speed). For such a set up, the maximum relative differences in the changes of propagation delay values can be accurately characterized. Since the maximum and minimum change in the propagation delay values are known, we can use the robust scheduling algorithm proposed in Chapter 5 to compute an optimal schedule that will be robust to the changes due to mobile nodes.

*Mobile network:* Although, Chapter 5 deals with limited mobility, it is not an approach to be followed if the setup is not that of a static network but rather is of a mobile network. In case of mobile network where the nodes can be Autonomous Underwater Vehicles (AUVs), a distributed solution seems to be a good approach and is subject of future work.

#### **Unknown traffic demands**

The centralized algorithm proposed in Chapter 4 has three inputs (i) links to be scheduled, (ii) the packet traffic demand for each link and (iii) the propagation delay information. Given these inputs, the algorithm finds an optimal throughput-maximizing schedule. In case of unknown traffic demand, the solution to the proposed algorithm will be sub-optimal. To design a policy based on unknown traffic demands might need a distributed approach to the problem with dynamic updates to the schedule.

To handle unknown traffic demands, the queue length of implemented queue at each node for arrival packets must be taken into consideration. Apart from minimizing the idle time which maximizes the throughput, an additional cost must be added to nodes with greater queue lengths while updating the schedule

every frame based on the traffic demand. The latency and stability of the network are more important parameters to consider in the case the traffic is not deterministic. This will result in a schedule that exploits large propagation delays and at the same time satisfies the unknown traffic demand.

The extension of the work presented here to random access networks is interesting. In ad-hoc networks, where nodes are added and deleted randomly, efficient topology discovery mechanisms, along with optimized contention-based MAC protocols leveraging on propagation delay information, is a subject that merits further investigation.

### **Unknown network topology**

The results presented in this thesis have shown that knowing the network topology can lead to MAC protocols with better performance. Therefore, in the absence of network topological information, an efficient mechanism that can be used to discover the network topology can be useful and is a pertinent subject for further research.

### **Practicality**

For static UWA networks with known propagation delays, we have demonstrated in this thesis that the techniques proposed are implementable and that they work in reality. It would be useful to integrate such methods into the existing protocol stacks available for UWA networks so as to provide more opportunities for the simulation and testing of such MAC protocols.

## Bibliography

---

- [1] M. Chitre, M. Motani, and S. Shahabudeen, "Throughput of networks with large propagation delays," *Oceanic Engineering, IEEE Journal of*, vol. 37, no. 4, pp. 645–658, 2012.
- [2] C.-C. Hsu, M.-S. Kuo, C.-F. Chou, and K. C.-J. Lin, "The elimination of spatial-temporal uncertainty in underwater sensor networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 21, no. 4, pp. 1229–1242, 2013.
- [3] S. Lmai, M. Chitre, C. Laot, and S. Houcke, "Throughput-maximizing transmission schedules for underwater acoustic multihop grid networks," *IEEE Journal of Oceanic Engineering*, vol. 40, no. 4, pp. 853–863, 2015.
- [4] A. Koulakezian, R. Ohannessian, H. Denkilkian, M. Chalfoun, M. K. Joujou, A. Chehab, and I. H. Elhajj, "Wireless sensor node for real-time thickness measurement and localization of oil spills," in *2008 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2008, pp. 631–636.
- [5] A. Faustine, A. N. Mvuma, H. J. Mongi, M. C. Gabriel, A. J. Tenge, and S. B. Kucel, "Wireless sensor networks for water quality monitoring and control within lake victoria basin: prototype development," *Wireless Sensor Network*, vol. 6, no. 12, p. 281, 2014.
- [6] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. ACM, 2002, pp. 88–97.
- [7] N. F. Henry and O. N. Henry, "Wireless sensor networks based pipeline vandalism and oil spillage monitoring and detection: main benefits for nigeria oil and gas sectors," *The SIJ Transactions on Computer Science Engineering & its Applications (CSEA)*, vol. 3, no. 1, pp. 1–6, 2015.
- [8] C. Allen, "Wireless sensor network nodes: security and deployment in the niger-delta oil and gas sector," 2011.
- [9] K. U. Menon, P. Divya, and M. V. Ramesh, "Wireless sensor network for river water quality monitoring in india," in *Computing Communication & Networking Technologies (ICCCNT), 2012 Third International Conference on*. IEEE, 2012, pp. 1–7.
- [10] M. López, S. Martinez, J. Gomez, A. Herms, L. Tort, J. Bausells, and A. Errachid, "Wireless monitoring of the ph, nh 4+ and temperature in a fish farm," *Procedia Chemistry*, vol. 1, no. 1, pp. 445–448, 2009.
- [11] C. Rao, K. Mukherjee, S. Gupta, A. Ray, and S. Phoha, "Underwater mine detection using symbolic pattern analysis of sidescan sonar images," in *2009 American Control Conference*. IEEE, 2009, pp. 5416–5421.
- [12] P. Kumar, P. Kumar, P. Priyadarshini *et al.*, "Underwater acoustic sensor network for early warning generation," in *2012 Oceans*. IEEE, 2012, pp. 1–6.
- [13] A. Khan and L. Jenkins, "Undersea wireless sensor network for ocean pollution prevention," in *Communication Systems Software and Middleware and Workshops, 2008. COMSWARE 2008. 3rd International Conference on*. IEEE, 2008, pp. 2–8.

- 
- [14] S. Zhou and P. Willett, "Submarine location estimation via a network of detection-only sensors," *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 3104–3115, 2007.
- [15] N. Mohamed, I. Jawhar, J. Al-Jaroodi, and L. Zhang, "Sensor network architectures for monitoring underwater pipelines," *Sensors*, vol. 11, no. 11, pp. 10 738–10 764, 2011.
- [16] C. Barngrover, R. Kastner, and S. Belongie, "Semisynthetic versus real-world sonar training data for the classification of mine-like objects," *IEEE Journal of Oceanic Engineering*, vol. 40, no. 1, pp. 48–56, 2015.
- [17] J. Trevathan, R. Johnstone, T. Chiffings, I. Atkinson, N. Bergmann, W. Read, S. Theiss, T. Myers, and T. Stevens, "Semat—the next generation of inexpensive marine environmental monitoring and measurement systems," *Sensors*, vol. 12, no. 7, pp. 9711–9748, 2012.
- [18] L. A. Abdul-Rahaim and A. M. A. Ali, "Remote wireless automation and monitoring of large farm using wireless sensors networks and internet," *International Journal of Computer Science & Engineering Technology*, vol. 6, no. 3, pp. 118–137, 2015.
- [19] H. Saeed, S. Ali, S. Rashid, S. Qaisar, and E. Felemban, "Reliable monitoring of oil and gas pipelines using wireless sensor network (wsn)—remong," in *System of Systems Engineering (SOSE), 2014 9th International Conference on*. IEEE, 2014, pp. 230–235.
- [20] S. Kumar, A. Perry, C. Moeller, D. Skvoretz, M. Ebbert, R. Ostrom, S. Bennett, and P. Czipott, "Real-time tracking magnetic gradiometer for underwater mine detection," in *OCEANS'04. MTS/IEEE TECHNO-OCEAN'04*, vol. 2. IEEE, 2004, pp. 874–878.
- [21] A. Pirisi, F. Grimaccia, M. Mussetta, R. E. Zich, R. Johnstone, M. Palaniswami, and S. Rajasegarar, "Optimization of an energy harvesting buoy for coral reef monitoring," in *2013 IEEE Congress on Evolutionary Computation*. IEEE, 2013, pp. 629–634.
- [22] D. P. Williams, "On optimal auv track-spacing for underwater mine detection," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4755–4762.
- [23] S. Zhang, J. Yu, A. Zhang, L. Yang, and Y. Shu, "Marine vehicle sensor network architecture and protocol designs for ocean observation," *Sensors*, vol. 12, no. 1, pp. 373–390, 2012.
- [24] A. Cerpa, J. Elson, D. Estrin, L. Girod, M. Hamilton, and J. Zhao, "Habitat monitoring: Application driver for wireless communications technology," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 2 supplement, pp. 20–41, 2001.
- [25] J. Lloret, S. Sendra, M. Garcia, and G. Lloret, "Group-based underwater wireless sensor network for marine fish farms," in *2011 IEEE GLOBECOM Workshops (GC Wkshps)*. IEEE, 2011, pp. 115–119.
- [26] E. N. Udo and E. B. Isong, "Flood monitoring and detection system using wireless sensor network," *Asian Journal of Computer and Information Systems*, vol. 1, 2013.
- [27] A. Pasi and U. Bhawe, "Flood detection system using wireless sensor network," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 2, pp. 386–389, 2015.

## BIBLIOGRAPHY

---

- [28] A. Yalcuk and S. Postalcioglu, "Evaluation of pool water quality of trout farms by fuzzy logic: monitoring of pool water quality for trout farms," *International Journal of Environmental Science and Technology*, vol. 12, no. 5, pp. 1503–1514, 2015.
- [29] S. Khaledi, H. Mann, J. Perkovich, and S. Zayed, "Design of an underwater mine detection system," in *Systems and Information Engineering Design Symposium (SIEDS), 2014*. IEEE, 2014, pp. 78–83.
- [30] N. Nasser, A. Zaman, L. Karim, and N. Khan, "Cpws: An efficient routing protocol for rgb sensor-based fish pond monitoring system," in *2012 IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2012, pp. 7–11.
- [31] G. Tuna, O. Arkoc, and K. Gulez, "Continuous monitoring of water quality using portable and low-cost approaches," *International Journal of Distributed Sensor Networks*, vol. 2013, 2013.
- [32] J. Kong, J.-h. Cui, D. Wu, and M. Gerla, "Building underwater ad-hoc networks and sensor networks for large scale real-time aquatic applications," in *Military Communications Conference, 2005. MILCOM 2005. IEEE*. IEEE, 2005, pp. 1535–1541.
- [33] S. Tyan and S.-H. Oh, "Auv-rm: underwater sensor network scheme for auv based river monitoring," *Research Trend in Computer and Applications, SERCE*, vol. 24, pp. 53–55, 2013.
- [34] H. Yang, H. Wu, and Y. He, "Architecture of wireless sensor network for monitoring aquatic environment of marine shellfish," in *Asian Control Conference, 2009. ASCC 2009. 7th*. IEEE, 2009, pp. 1147–1151.
- [35] M. J. Hamilton, S. Kemna, and D. Hughes, "Antisubmarine warfare applications for autonomous underwater vehicles: the glint09 sea trial results," *Journal of Field Robotics*, vol. 27, no. 6, pp. 890–902, 2010.
- [36] H. Yu and M. Guo, "An efficient oil and gas pipeline monitoring systems based on wireless sensor networks," in *Information Security and Intelligence Control (ISIC), 2012 International Conference on*. IEEE, 2012, pp. 178–181.
- [37] G. Acar and A. Adams, "Acmenet: an underwater acoustic sensor network protocol for real-time environmental monitoring in coastal areas," *IEEE Proceedings-Radar, Sonar and Navigation*, vol. 153, no. 4, pp. 365–380, 2006.
- [38] C. A. Pérez, M. Jimenez, F. Soto, R. Torres, J. López, and A. Iborra, "A system for monitoring marine environments based on wireless sensor networks," in *OCEANS 2011 IEEE-Spain*. IEEE, 2011, pp. 1–6.
- [39] L. Freitag, M. Grund, C. Von Alt, R. Stokey, and T. Austin, "A shallow water acoustic network for mine countermeasures operations with autonomous underwater vehicles," *Underwater Defense Technology (UDT)*, pp. 1–6, 2005.
- [40] K. Casey, A. Lim, and G. Dozier, "A sensor network architecture for tsunami detection and response," *International Journal of Distributed Sensor Networks*, vol. 4, no. 1, pp. 28–43, 2008.
- [41] C. Alippi, R. Camplani, C. Galperti, and M. Roveri, "A robust, adaptive, solar-powered wsn framework for aquatic environmental monitoring," *IEEE Sensors Journal*, vol. 11, no. 1, pp. 45–55, 2011.

- [42] R. Marin-Perez, J. García-Pintado, and A. S. Gómez, “A real-time measurement system for long-life flood monitoring and warning applications,” *Sensors*, vol. 12, no. 4, pp. 4213–4236, 2012.
- [43] I. Jawhar, N. Mohamed, and K. Shuaib, “A framework for pipeline infrastructure monitoring using wireless sensor networks,” in *Wireless Telecommunications Symposium, 2007. WTS 2007*. IEEE, 2007, pp. 1–7.
- [44] R. Been, D. T. Hughes, and A. Vermeij, “Heterogeneous underwater networks for asw: technology and techniques,” *Proc. of Underwater Defence Technology (UDT)*, 2008.
- [45] R. Headrick and L. Freitag, “Growth of underwater communication technology in the us navy,” *IEEE Communications Magazine*, vol. 47, no. 1, pp. 80–82, 2009.
- [46] U. M. Cella, R. Johnstone, and N. Shuley, “Electromagnetic wave wireless communication in shallow water coastal environment: theoretical analysis and experimental results,” in *Proceedings of the Fourth ACM International Workshop on UnderWater Networks*. ACM, 2009, p. 9.
- [47] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke, “Data collection, storage, and retrieval with an underwater sensor network,” in *Proceedings of the 3rd international conference on Embedded networked sensor systems*. ACM, 2005, pp. 154–165.
- [48] N. Farr, A. Bowen, J. Ware, C. Pontbriand, and M. Tivey, “An integrated, underwater optical/acoustic communications system,” in *OCEANS 2010 IEEE-Sydney*. IEEE, 2010, pp. 1–6.
- [49] T. Melodia, H. Kulhandjian, L.-C. Kuo, and E. Demirors, “Advances in underwater acoustic networking,” *Mobile ad hoc networking: Cutting edge directions*, vol. 852, 2013.
- [50] R. J. Urick, *Principles of underwater sound for engineers*. Tata McGraw-Hill Education, 1967.
- [51] M. Stojanovic, “On the relationship between capacity and distance in an underwater acoustic communication channel,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 34–43, 2007.
- [52] J. Heidemann, W. Ye, J. Wills, A. Syed, and Y. Li, “Research challenges and applications for underwater sensor networking,” in *Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE*, vol. 1. IEEE, 2006, pp. 228–235.
- [53] J. Rice, B. Creber, C. Fletcher, P. Baxley, K. Rogers, K. McDonald, D. Rees, M. Wolf, S. Merriam, R. Mehio *et al.*, “Evolution of seaweb underwater acoustic networking,” in *Oceans 2000 MTS/IEEE Conference and Exhibition*, vol. 3. IEEE, 2000, pp. 2007–2017.
- [54] D. Pompili, T. Melodia, and I. F. Akyildiz, “A cdma-based medium access control for underwater acoustic sensor networks,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 4, pp. 1899–1909, 2009.
- [55] J. Heidemann, M. Stojanovic, and M. Zorzi, “Underwater sensor networks: applications, advances and challenges,” *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 370, no. 1958, pp. 158–175, 2012.

## BIBLIOGRAPHY

---

- [56] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad hoc networks*, vol. 3, no. 3, pp. 257–279, 2005.
- [57] M. Chitre, S. Shahabudeen, and M. Stojanovic, "Underwater acoustic communications and networking: Recent advances and future challenges," *Marine technology society journal*, vol. 42, no. 1, pp. 103–116, 2008.
- [58] M. K. Park and V. Rodoplu, "Uwan-mac: an energy-efficient mac protocol for underwater acoustic wireless sensor networks," *IEEE journal of oceanic engineering*, vol. 32, no. 3, pp. 710–720, 2007.
- [59] P. Casari, F. E. Lapicciarella, and M. Zorzi, "A detailed simulation study of the uwan-mac protocol for underwater acoustic networks," in *OCEANS 2007*. IEEE, 2007, pp. 1–6.
- [60] Y. Zhong, J. Huang, and J. Han, "A tdma mac protocol for underwater acoustic sensor networks," in *Information, Computing and Telecommunication, 2009. YC-ICT'09. IEEE Youth Conference on*. IEEE, 2009, pp. 534–537.
- [61] C. Yun, A.-R. Cho, S.-G. Kim, J.-W. Park, and Y.-K. Lim, "A hierarchical time division multiple access medium access control protocol for clustered underwater acoustic networks," *Journal of information and communication convergence engineering*, vol. 11, no. 3, pp. 153–166, 2013.
- [62] Z. Li, Z. Guo, H. Qu, F. Hong, P. Chen, and M. Yang, "Ud-tdma: a distributed tdma protocol for underwater acoustic sensor network," in *2009 IEEE 6th International Conference on Mobile Adhoc and Sensor Systems*. IEEE, 2009, pp. 918–923.
- [63] P.-H. Huang, Y. Chen, B. Krishnamachari, and A. Kumar, "Link scheduling in a single broadcast domain underwater networks," in *Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), 2010 IEEE International Conference on*. IEEE, 2010, pp. 205–212.
- [64] D. Marinakis, K. Wu, N. Ye, and S. Whitesides, "Network optimization for lightweight stochastic scheduling in underwater sensor networks," *IEEE Transactions on Wireless Communications*, vol. 11, no. 8, pp. 2786–2795, 2012.
- [65] X. Guo, M. R. Frater, and M. J. Ryan, "A propagation-delay-tolerant collision avoidance protocol for underwater acoustic sensor networks," in *OCEANS 2006-Asia Pacific*. IEEE, 2007, pp. 1–6.
- [66] P. Xie and J.-H. Cui, "R-mac: An energy-efficient mac protocol for underwater sensor networks," in *Wireless Algorithms, Systems and Applications, 2007. WASA 2007. International Conference on*. IEEE, 2007, pp. 187–198.
- [67] H.-H. Ng, W.-S. Soh, and M. Motani, "Maca-u: A media access protocol for underwater acoustic networks," in *IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference*. IEEE, 2008, pp. 1–5.
- [68] N. Chirdchoo, W.-s. Soh, and K. C. Chua, "Ript: A receiver-initiated reservation-based protocol for underwater acoustic networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 9, pp. 1744–1753, 2008.

- 
- [69] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "Maca-mn: A maca-based mac protocol for underwater acoustic networks with packet train for multiple neighbors," in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*. IEEE, 2008, pp. 46–50.
- [70] L. Badia, M. Mastrogiovanni, C. Petrioli, S. Stefanakos, and M. Zorzi, "An optimization framework for joint sensor deployment, link scheduling and routing in underwater sensor networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 44–56, 2007.
- [71] K. Kredo, P. Djukic, P. Mohapatra *et al.*, "STUMP: Exploiting position diversity in the staggered TDMA underwater MAC protocol," in *INFOCOM 2009, IEEE*. IEEE, 2009, pp. 2961–2965.
- [72] K. Kredo II and P. Mohapatra, "Distributed scheduling and routing in underwater wireless networks," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. IEEE, 2010, pp. 1–6.
- [73] J. Ma and W. Lou, "Interference-aware spatio-temporal link scheduling for long delay underwater sensor networks," in *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011 8th Annual IEEE Communications Society Conference on*. IEEE, 2011, pp. 431–439.
- [74] F. L. Presti, C. Petrioli, R. Petroccia, and A. Shashaj, "A scalable analytical framework for deriving optimum scheduling and routing in underwater sensor networks," in *Mobile Adhoc and Sensor Systems (MASS), 2012 IEEE 9th International Conference on*. IEEE, 2012, pp. 127–135.
- [75] H.-H. Ng, W.-S. Soh, and M. Motani, "A bidirectional-concurrent mac protocol with packet bursting for underwater acoustic networks," *IEEE Journal of Oceanic Engineering*, vol. 38, no. 3, pp. 547–565, 2013.
- [76] B. Hou, O. Hinton, A. Adams, and B. Sharif, "An time-domain-oriented multiple access protocol for underwater acoustic network communications," in *OCEANS'99 MTS/IEEE. Riding the Crest into the 21st Century*, vol. 2. IEEE, 1999, pp. 585–589.
- [77] V. R. Cadambe and S. A. Jafar, "Interference alignment and degrees of freedom of the user interference channel," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3425–3441, 2008.
- [78] —, "Degrees of freedom of wireless networks-what a difference delay makes," in *2007 Conference Record of the Forty-First Asilomar Conference on Signals, Systems and Computers*. IEEE, 2007, pp. 133–137.
- [79] R. Mathar and G. Bocherer, "On spatial patterns of transmitter-receiver pairs that allow for interference alignment by delay," in *Signal Processing and Communication Systems, 2009. ICSPCS 2009. 3rd International Conference on*. IEEE, 2009, pp. 1–5.
- [80] F. L. Blasco, F. Rossetto, and G. Bauch, "Time interference alignment via delay offset for long delay networks," *IEEE Transactions on Communications*, vol. 62, no. 2, pp. 590–599, 2014.
- [81] S. Lmai, M. Chitre, C. Laot, and S. Houcke, "Throughput-efficient super-tdma mac transmission schedules in ad hoc linear underwater acoustic networks."

## BIBLIOGRAPHY

---

- [82] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3. IEEE, 2002, pp. 1567–1576.
- [83] A. A. Syed, W. Ye, and J. Heidemann, "T-lohi: A new class of mac protocols for underwater acoustic sensor networks," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*. IEEE, 2008.
- [84] M. Cheng and L. Yin, "Transmission scheduling in sensor networks via directed edge coloring," in *2007 IEEE International Conference on Communications*. IEEE, 2007, pp. 3710–3715.
- [85] H. Zeng, Y. T. Hou, Y. Shi, W. Lou, S. Kompella, and S. F. Midkiff, "SHARK-IA: An interference alignment algorithm for multi-hop underwater acoustic networks with large propagation delays," in *Proceedings of the International Conference on Underwater Networks & Systems*. ACM, 2014, p. 6.
- [86] P. Anjani and M. Chitre, "Scheduling algorithm with transmission power control for random underwater acoustic networks," in *OCEANS 2015-Genova*. IEEE, 2015, pp. 1–8.
- [87] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *Information Theory, IEEE Transactions on*, vol. 46, no. 2, pp. 388–404, 2000.
- [88] J. Deng, B. Liang, and P. K. Varshney, "Tuning the carrier sensing range of ieee 802.11 mac," in *Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE*, vol. 5. IEEE, 2004, pp. 2987–2991.
- [89] S. Boppana and J. M. Shea, "Overlapped carrier-sense multiple access (ocsma) in wireless ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 3, pp. 369–383, 2009.
- [90] L. Busoniu, R. Babuska, B. De Schutter, and D. Ernst, *Reinforcement learning and dynamic programming using function approximators*. CRC press, 2010, vol. 39.
- [91] W. B. Powell, *Approximate Dynamic Programming: Solving the curses of dimensionality*. John Wiley & Sons, 2007, vol. 703.
- [92] P. Anjani and M. Chitre, "Propagation delay-aware unslotted scheduling with variable packet duration for underwater acoustic networks," *Oceanic Engineering, IEEE Journal of*, submitted for publication.
- [93] S. Lmai, M. Chitre, C. Laot, and S. Houcke, "Tdma-based mac transmission schedules in multihop grid ad hoc underwater acoustic networks," in *Underwater Communications and Networking (UComms), 2014*. IEEE, 2014, pp. 1–5.
- [94] S. Lee and I. E. Grossmann, "New algorithms for nonlinear generalized disjunctive programming," *Computers & Chemical Engineering*, vol. 24, no. 9, pp. 2125–2141, 2000.
- [95] I. E. Grossmann and F. Trespalacios, "Systematic modeling of discrete-continuous optimization models through generalized disjunctive programming," *AIChE Journal*, vol. 59, no. 9, pp. 3276–3295, 2013.
- [96] L. A. Wolsey and G. L. Nemhauser, *Integer and combinatorial optimization*. John Wiley & Sons, 2014.

- [97] D. Yue, G. Guillén-Gosálbez, and F. You, “Global optimization of large-scale mixed-integer linear fractional programming problems: A reformulation-linearization method and process scheduling applications,” *AIChE Journal*, vol. 59, no. 11, pp. 4255–4272, 2013.
- [98] Z. Zhong and F. You, “Parametric algorithms for global optimization of mixed-integer fractional programming problems in process engineering,” in *American Control Conference (ACC), 2014*. IEEE, 2014, pp. 3609–3614.
- [99] M. Chitre, I. Topor, and T.-B. Koay, “The UNET-2 modem—an extensible tool for underwater networking research,” in *OCEANS, 2012-Yeosu*. IEEE, 2012, pp. 1–7.
- [100] M. Chitre, I. Topor, R. Bhatnagar, and V. Pallayil, “Variability in link performance of an underwater acoustic network,” in *OCEANS-Bergen, 2013 MTS/IEEE*. IEEE, 2013, pp. 1–7.
- [101] P. Anjani and M. Chitre, “Unslotted transmission schedules for practical underwater acoustic multihop grid networks with large propagation delays,” in *UCOMMS 2016-Lerici*. IEEE, 2016, pp. 1–5.
- [102] —, “Unslotted and robust schedules for practical underwater acoustic multihop grid networks with propagation delay uncertainty,” *Oceanic Engineering, IEEE Journal of*, submitted for publication.
- [103] M. Stojanovic, “Capacity of a relay acoustic channel,” in *OCEANS 2007*. IEEE, 2007, pp. 1–7.
- [104] W. Zhang, M. Stojanovic, and U. Mitra, “Analysis of a linear multihop underwater acoustic network,” *IEEE Journal of Oceanic Engineering*, vol. 35, no. 4, pp. 961–970, 2010.
- [105] P. Anjani and M. Chitre, “Design and implementation of Super-TDMA: A mac protocol exploiting large propagation delays for underwater acoustic networks,” in *Proceedings of the 10th International Conference on Underwater Networks & Systems*, ser. WUWNET ’15. New York, NY, USA: ACM, 2015, pp. 1:1–1:8. [Online]. Available: <http://doi.acm.org/10.1145/2831296.2831299>
- [106] J. U. Robert, “Principles of underwater sound,” *New York: McGraw-Hill Book Company*, 1983.
- [107] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [108] “The Underwater Networks Project: 2014. <http://www.unetstack.net/doc/html/index.html>. Accessed: 2015-07-07.”
- [109] M. Chitre, R. Bhatnagar, and W.-S. Soh, “UnetStack: an agent-based software stack and simulator for underwater networks,” in *Oceans-St. John’s, 2014*. IEEE, 2014, pp. 1–10.
- [110] S. Shahabudeen, “Time domain medium access control protocols for underwater acoustic networks.” Ph.D. dissertation, Natl. Univ. of Singapore, 2012.
- [111] P. Anjani and M. Chitre, “Experimental demonstration of super-tdma: A mac protocol exploiting large propagation delays in underwater acoustic networks,” in *UCOMMS 2016-Lerici*. IEEE, 2016, pp. 1–5.

## BIBLIOGRAPHY

---

- [112] Y. Zhu, Z. Peng, J.-H. Cui, and H. Chen, "Toward practical MAC design for underwater acoustic networks," *Mobile Computing, IEEE Transactions on*, vol. 14, no. 4, pp. 872–886, 2015.
- [113] 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 *OCEANS, 2005. Proceedings of MTS/IEEE*. IEEE, 2005, pp. 1086–1092.
- [114] H. Yan, L. Wan, S. Zhou, Z. Shi, J.-H. Cui, J. Huang, and H. Zhou, "DSP based receiver implementation for OFDM acoustic modems," *Physical Communication*, vol. 5, no. 1, pp. 22–32, 2012.

## List of Publications

---

### Journals

- (J1) P. Anjangi, M. Chitre, "Propagation Delay-Aware Unslotted Schedules with Variable Packet Duration for Underwater Acoustic Networks," *IEEE Journal of Oceanic Engineering*, (Accepted).
- (J2) P. Anjangi, M. Chitre, "Unslotted & Robust Schedules for Practical Underwater Acoustic Multihop Grid Networks with Propagation Delay Uncertainty," *IEEE Journal of Oceanic Engineering*, (Under review).

### Conferences

- (C1) P. Anjangi, M. Chitre, "Unslotted Transmission Schedules for Practical Underwater Acoustic Multihop Grid Networks with Large Propagation Delays," *Proceedings of IEEE UComms*, Lerici, Italy, Aug 30-Sep 1, 2016.
- (C2) P. Anjangi, M. Chitre, "Experimental Demonstration of Super-TDMA: A MAC Protocol Exploiting Large Propagation Delays in Underwater Acoustic Networks," *Proceedings of IEEE UComms*, Lerici, Italy, Aug 30-Sep 1, 2016.
- (C3) P. Anjangi, M. Chitre, "Design and Implementation of Super-TDMA: A MAC Protocol Exploiting Large Propagation Delays for Underwater Acoustic Networks," *Proceedings of 10th ACM International Conf. on Underwater Networks & Systems (WUWNet)*, Washington DC, Oct 22-24, 2015, (**Best Experimental Student Paper Award**).
- (C4) P. Anjangi, M. Chitre, "Scheduling Algorithm with Transmission Power Control for Random Underwater Acoustic Networks," *Proceedings of IEEE Oceans 2015*, Genoa, Italy, May 18-21, 2015.