

# **Data Acquisition and Dissemination for the Internet of Nano Things**

by

Hang Yu

A thesis  
submitted to the Victoria University of Wellington  
in fulfilment of the  
requirements for the degree of  
Doctor of Philosophy  
in Engineering.

Victoria University of Wellington  
2018



## Abstract

Electromagnetic-based Wireless NanoSensor Networks (EM-WNSNs) operating in the Terahertz band (0.1 THz – 10 THz) will enable nano-scale applications and stimulate the evolution from the Internet of Things (IoT) to the Internet of Nano Things (IoNT). Data delivery, which is one of the key functionalities of EM-WNSNs, faces three major challenges that will affect network performance: the frequency-selective channel in the THz band due to molecular absorption, the limited ability to support networking functions due to their small size, and the limited bandwidth of the existing infrastructure for transferring sensed data to the Internet. However, to date, limited amount of research on data delivery has been done to address the peculiarities of IoNT from the networking perspective.

To fill the gap, in this thesis, data acquisition and dissemination solutions are studied for IoNT to improve the resource utilization efficiency during data delivery. Different from existing literatures that focus on standalone nanonetworks, this thesis investigates solutions for connecting nano-devices to the Internet.

In detail, the contributions of this thesis are composed of four components: first, a preliminary study namely the Channel-aware Forwarding (CForward) is conducted on multi-hop forwarding for THz networks; second, the On-demand Probabilistic polling (OP polling) is developed for IoNT with dynamic IoT bandwidth and channel conditions; third, a TTL-based Efficient Forwarding (TEForward) is designed for the polling-based nanonetworks under dynamic channel conditions; fourth, the Enhanced Adaptive Pulse Interval Scheduling (EAPIS) is implemented to collect data from event-based nanonetworks under limited IoT bandwidth.



# Acknowledgments

Thank my supervisors, Professor Winston K.G. Seah and Dr Bryan Ng, for their endless support in every aspect during this long journey. Also thank my wife for her unconditional sacrifice and support. Besides, thank my parents for their spiritual and financial investments on me. Last but not the least, this thesis cannot be accomplished without the help from all my research colleagues and friends.



# Awards and Publications

## Awards

- VUW Postgraduate Research Excellence Awards 2017
- IEEE LCN 2017 Best Paper Award

## Publications

### Journal

The following journal article is derived from Chapter 4

- Hang Yu, Bryan Ng and Winston K.G. Seah, On-demand Probabilistic Polling for Nanonetworks under Dynamic IoT Backhaul Network Conditions. *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 2217-2227, Dec. 2017. [1]

The following journal manuscript derived from Chapter 5 is under review:

- Hang Yu, Bryan Ng and Winston K.G. Seah, TTL-based Efficient Forwarding for Nanonetworks with Multiple Coordinated IoT Gateways. Submitted to *IEEE Internet of Things Journal*, December 2017.

The following journal manuscript derived from Chapter 6 is under review:

- Hang Yu, Bryan Ng and Winston K.G. Seah, Enhanced Pulse Arrival Scheduling for Nanonetworks under Limited IoT Access Bandwidth. Submitted to *IEEE Internet of Things Journal*, February 2018.

## Conference

The following conference paper is derived from Chapter 6:

- Hang Yu, Bryan Ng and Winston K.G. Seah, Pulse Arrival Scheduling for Nanonetworks under Limited IoT Access Bandwidth, Best Paper Award, *Proceedings of the 42nd IEEE Conference on Local Computer Networks (LCN)*, October 9-12, 2017, Singapore. [2]

The following conference paper is the early work of Chapter 5:

- Hang Yu, Bryan Ng, Winston K.G. Seah and Ying Qu, TTL-based Efficient Forwarding for the Backhaul tier in Nanonetworks. *Proceedings of the 14th Annual IEEE Consumer Communications & Networking Conference (CCNC)*, 8-11 January 2017, Las Vegas, USA. [3]

The following conference paper originates from Chapter 4:

- Hang Yu, Bryan Ng and Winston K.G. Seah, On-demand Efficient Polling for Nanonetworks under Dynamic IoT Backhaul Network Conditions. *Proceedings of the 35rd IEEE International Performance Computing and Communications Conference (IPCCC)*, 9-11 December 2016, Las Vegas, USA. [4]

The following conference paper is a preliminary work presented in Chapter 3:

- Hang Yu, Bryan Ng and Winston K.G. Seah, Forwarding Schemes for EM-based Wireless NanoSensor Networks in the Terahertz Band. *Proceedings of the 2nd ACM International Conference on Nanoscale Computing and Communication (NANOCOM)*, Boston, MA, USA, 21-22 Sep 2015. [5]

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Research Challenges . . . . .	3
1.2	Contributions . . . . .	4
1.3	Organisation of Thesis . . . . .	7
<b>2</b>	<b>Background and Related Work</b>	<b>9</b>
2.1	IoNT: Overview . . . . .	9
2.1.1	Graphene-based Nano Technology . . . . .	11
2.1.2	THz Channel . . . . .	12
2.1.3	Nano-device and Network Architecture . . . . .	17
2.1.4	IoT Access and Backhaul Solutions . . . . .	20
2.1.5	IoNT Applications . . . . .	21
2.2	IoNT: MAC Layer . . . . .	23
2.2.1	Frequency Allocation . . . . .	24
2.2.2	Time Allocation . . . . .	25
2.2.3	Summary of MAC Layer in IoNT . . . . .	30
2.3	IoNT: Network Layer . . . . .	32
2.3.1	Centralized Routing in IoNT . . . . .	32
2.3.2	Distributed Routing in IoNT . . . . .	34
2.3.3	Summary of Network Layer in IoNT . . . . .	37
2.4	Polling in WSNs . . . . .	38
2.4.1	Polling in Battery-powered WSNs . . . . .	39
2.4.2	Polling in Energy-harvesting WSNs . . . . .	39

2.4.3	Summary of Polling in WSNs . . . . .	40
2.5	Summary . . . . .	42
<b>3</b>	<b>Preliminary Study: CForward</b>	<b>43</b>
3.1	Algorithm Design . . . . .	43
3.2	Performance Evaluation . . . . .	45
3.2.1	Performance Metrics . . . . .	45
3.2.2	Benchmarks . . . . .	46
3.2.3	Simulation Setup . . . . .	47
3.2.4	Varying Node Densities . . . . .	48
3.2.5	Varying H <sub>2</sub> O Percentage . . . . .	50
3.2.6	Discussion on the Optimal Case . . . . .	52
3.3	Summary . . . . .	53
<b>4</b>	<b>Bandwidth-aware Polling for IoNT</b>	<b>54</b>
4.1	Optimal Packet Size for Polling . . . . .	55
4.2	On-demand Probabilistic Polling . . . . .	57
4.2.1	Assumptions . . . . .	58
4.2.2	Algorithm Design . . . . .	58
4.3	Performance Evaluation . . . . .	63
4.3.1	Performance Metrics . . . . .	63
4.3.2	Benchmarks . . . . .	65
4.3.3	Data Dissemination for the Polling System . . . . .	67
4.3.4	Simulation Parameters . . . . .	68
4.3.5	Results and Analysis . . . . .	71
4.4	Summary . . . . .	82
<b>5</b>	<b>Forwarding for Polling-based IoNT</b>	<b>83</b>
5.1	Optimal Energy-efficient Forwarding . . . . .	84
5.2	TTL-based Efficient Forwarding . . . . .	85
5.3	Performance Evaluation . . . . .	92
5.3.1	Performance Metrics . . . . .	92

5.3.2	Benchmarks . . . . .	94
5.3.3	Simulation Setup . . . . .	95
5.3.4	Results and Analysis . . . . .	97
5.4	Summary . . . . .	112
<b>6</b>	<b>Bandwidth-aware Access for Event-driven IoNT</b>	<b>113</b>
6.1	Enhanced Adaptive Pulse Interval . . . . .	113
6.1.1	Bursty Events and Ideal Pulse Arrival . . . . .	114
6.1.2	Overview of EAPIS . . . . .	115
6.1.3	EAPIS Preamble . . . . .	116
6.1.4	Transmission Sequencing . . . . .	117
6.1.5	EAPIS Example . . . . .	120
6.2	Performance Evaluation . . . . .	121
6.2.1	Performance Metrics . . . . .	121
6.2.2	Benchmarks . . . . .	123
6.2.3	Simulation Parameters . . . . .	128
6.2.4	Results and Analysis . . . . .	129
6.3	Performance Modelling . . . . .	136
6.3.1	EAPIS Bandwidth Efficiency . . . . .	136
6.3.2	APIS Bandwidth Efficiency . . . . .	138
6.3.3	EAPIS Unit Energy Consumption . . . . .	139
6.3.4	APIS Unit Energy Consumption . . . . .	143
6.4	Summary . . . . .	144
<b>7</b>	<b>Conclusions and Future Work</b>	<b>145</b>
7.1	Review . . . . .	145
7.2	Contributions . . . . .	146
7.3	Future Work . . . . .	146

# List of Figures

2.1	Communication options for wireless nanosensor networks [6]	10
2.2	Molecular communications [7]	10
2.3	Structure of graphene [8]	11
2.4	The path loss in the THz band (USA Model, Mean Latitude, Summer, H=0 [9])	13
2.5	TS-OOK modulation	16
2.6	An integrated nano-device [6]	17
2.7	Overall network architecture of IoNT	19
3.1	Network topology with ten nano nodes	47
3.2	Performance for uniform topology with 10% H <sub>2</sub> O	48
3.3	Performance for random topology with 10% H <sub>2</sub> O	49
3.4	Performance for uniform topology with varying H <sub>2</sub> O	51
3.5	Performance for random topology with varying H <sub>2</sub> O	51
3.6	Comparison between CForward and the optimal	52
4.1	EM-WNSNs bridged to the IoT gateway	54
4.2	Sequence of message exchange in OP polling	59
4.3	Packet structure of beacons	60
4.4	Sequence of message exchange in OE polling	66
4.5	Packet structure of beacons in OE polling	66
4.6	IoT bandwidth capacity ratio and channel state vs time	71
4.7	Packet aggregation size for LSV channel	71

4.8	Packet aggregation size for SSV channel . . . . .	72
4.9	Bandwidth efficiency for LSV channel . . . . .	73
4.10	Bandwidth efficiency for SSV channel . . . . .	73
4.11	Packet drop ratio for LSV channel . . . . .	74
4.12	Packet drop ratio for SSV channel . . . . .	74
4.13	Energy consumption for LSV channel . . . . .	75
4.14	Energy consumption for SSV channel . . . . .	75
4.15	Polling coverage for LSV channel . . . . .	76
4.16	Polling coverage for SSV channel . . . . .	77
4.17	Polling delay for LSV channel . . . . .	78
4.18	Polling delay for SSV channel . . . . .	78
4.19	Collision probability for LSV channel . . . . .	79
4.20	Collision probability for SSV channel . . . . .	79
4.21	Fairness index for LSV channel . . . . .	80
4.22	Fairness index for SSV channel . . . . .	81
5.1	Structure of a polling beacon. . . . .	87
5.2	TEForward example for a single-gateway network. . . . .	90
5.3	TEForward example for a multi-gateway network. . . . .	91
5.4	$T_{wait}$ vs time . . . . .	96
5.5	Packet delivery ratio for SGW with LSV channel . . . . .	97
5.6	Packet delivery ratio for SGW with SSV channel . . . . .	97
5.7	Packet delivery ratio for MGW with LSV channel . . . . .	98
5.8	Packet delivery ratio for MGW with SSV channel . . . . .	98
5.9	Collision probability for SGW with LSV channel . . . . .	100
5.10	Collision probability for SGW with SSV channel . . . . .	100
5.11	Collision probability for MGW with LSV channel . . . . .	100
5.12	Collision probability for MGW with SSV channel . . . . .	101
5.13	Delay for SGW with LSV channel . . . . .	101
5.14	Delay for SGW with SSV channel . . . . .	101
5.15	Delay for MGW with LSV channel . . . . .	102
5.16	Delay for MGW with SSV channel . . . . .	102

5.17	Energy consumption for SGW with LSV channel . . . . .	103
5.18	Energy consumption for SGW with SSV channel . . . . .	104
5.19	Energy consumption for MGW with LSV channel . . . . .	104
5.20	Energy consumption for MGW with SSV channel . . . . .	104
5.21	Forwarder count for SGW with LSV channel . . . . .	105
5.22	Forwarder count for SGW with SSV channel . . . . .	106
5.23	Forwarder count for MGW with LSV channel . . . . .	106
5.24	Forwarder count for MGW with SSV channel . . . . .	106
5.25	Cumulative receiver count for SGW with LSV channel . . . .	107
5.26	Cumulative receiver count for SGW with SSV channel . . . .	107
5.27	Cumulative receiver count for MGW with LSV channel . . . .	107
5.28	Cumulative receiver count for MGW with SSV channel . . . .	108
5.29	Fairness for SGW with LSV channel . . . . .	108
5.30	Fairness for SGW with SSV channel . . . . .	109
5.31	Fairness for MGW with LSV channel . . . . .	109
5.32	Fairness for MGW with SSV channel . . . . .	109
6.1	Ideal pulse arrivals at the gateway . . . . .	115
6.2	Packet structure . . . . .	116
6.3	Adaptive pulse interval Scheduling ( $N' = 3, N = 3$ ) . . . . .	121
6.4	Adaptive pulse interval Scheduling ( $N' = 2, N = 3$ ) . . . . .	126
6.5	Raw throughput over time ( $N = 20$ ) . . . . .	130
6.6	Bandwidth efficiency vs network size . . . . .	130
6.7	Pulse drop ratio vs network size . . . . .	131
6.8	Dropped pulse ID vs network size . . . . .	131
6.9	Unit energy consumption vs network size . . . . .	133
6.10	First packet delay vs network size . . . . .	134
6.11	Pulse delay vs network size . . . . .	134
6.12	Fairness vs network size . . . . .	135
6.13	Packet delivery ratio vs network size . . . . .	136
6.14	Bandwidth efficiency modelling for EAPIS (Model: blue surface; Simulation: red surface) . . . . .	138

6.15	Bandwidth efficiency modelling for APIS (Model: blue surface; Simulation: red surface) . . . . .	139
6.16	Unit energy consumption modelling for EAPIS (Model: blue surface; Simulation: red surface) . . . . .	142
6.17	Unit energy consumption modelling for APIS (Model: blue surface; Simulation: red surface) . . . . .	143

# List of Tables

1.1	Contribution summary . . . . .	5
2.1	Summary of MAC (Frequency Allocation) Protocols in EM-WNSNs . . . . .	30
2.2	Summary of MAC (time allocation) protocols in EM-WNSNs	31
2.3	Summary of routing protocols in EM-WNSNs . . . . .	37
2.4	Summary of Polling in WSNs . . . . .	41
4.1	Notations used to describe OP polling . . . . .	60
4.2	Simulation parameters . . . . .	69
4.3	LSV-channel performance summary . . . . .	81
4.4	SSV-channel performance summary . . . . .	82
5.1	Notations used to describe TEForward . . . . .	86
5.2	Simulation parameters . . . . .	96
5.3	Single-gateway LSV-channel performance summary . . . . .	110
5.4	Single-gateway SSV-channel performance summary . . . . .	110
5.5	Multi-gateway LSV-channel performance summary . . . . .	111
5.6	Multi-gateway SSV-channel performance summary . . . . .	111
6.1	Notations for EAPIS of Nano-sink $k$ . . . . .	118
6.2	Simulation parameters . . . . .	128
6.3	Performance summary . . . . .	136

# Chapter 1

## Introduction

Driven by the rapid development of the nano materials such as graphene-based technologies, Electromagnetic-based Wireless NanoSensor Networks (EM-WNSNs) operating in the TeraHertz (THz) band (0.1 THz–10 THz) extend the sensing capabilities of Wireless Sensor Networks (WSNs) to the nano scale and stimulate the evolution from the Internet of Things (IoT) to the Internet of Nano Things (IoNT) [10]. The general network architecture of IoNT is composed of nanosensors, nano-sinks (nano-routers), macro-scale interface devices and IoT gateways that connect nano-devices to the Internet [6].

Nanosensors are responsible for event detection and propagation. Benefiting from the intrinsic properties of graphene, nanosensors in EM-WNSNs are endowed with high sensitivity and small size. In comparison, nano-sinks with relatively large size aim to coordinate sensors and aggregate the sensed data. Besides the benefits of advanced nano materials, nano-scale communications exhibit high throughput, energy efficiency, and reliability because of two reasons: 1) graphene-based antennas readily support pulse-based communications in the THz band [11] that supplies rich bandwidth resource; 2) the ultra high node density that increase network robustness. Owing to the above-mentioned characteristics, EM-WNSNs are capable of performing non-invasive tasks with a high sensing resolution

that cannot be accomplished by traditional macro-scale sensor networks. The insights extracted from nano-scale sensing benefit a wide range of applications such as inner / intra body health monitoring, drug delivery, pest monitoring, biochemical weapon detection, high-speed communications between computing cores, and smart materials [6] [12].

One of the most significant goals of sensor networks is data delivery. For EM-WNSNs in the context of IoT, transferring data from nanosensors to the Internet faces the challenges from both nano-scale networks and the existing macro-scale infrastructures. First, THz links are sensitive to molecular absorption that is determined by the molecular composition of communication media. Therefore, the transmission range is not only short but also dynamic due to the high frequency and molecular absorption. Next, the low capacity (e.g. energy, compute, storage, communication, etc.) of nano-devices limits the application of advanced networking solutions with high complexity. Networking solutions for nanonetworks should be lightweight. Last but not the least, the high throughput of nanonetworks is constrained by the limited and dynamic access and backhaul bandwidth of the existing macro-scale systems. In this thesis, the bandwidth and link of the IoT access and backhaul network are collectively referred to as the IoT bandwidth and IoT link, respectively. Cross-layer efforts are needed to adjust the behaviour of nanonetworks to match the existing infrastructure. The factors above introduce cross-layer constraints that reduce resource utilization efficiency of the overall IoNT during data delivery process.

To implement efficient end-to-end data delivery for IoNT, lightweight and fine-grained techniques in multiple layers of the protocol stack need to be designed. To date, a substantial amount of research from the physical layer to the network layer [6, 13–26] has been conducted. Nevertheless, most of the research outputs focus on communication mechanisms

for standalone nanonetworks under static environment. Networking solutions that comprehensively take into consideration the above-mentioned research challenges remain sparse. To fill the existing research gap, this thesis implements efficient data delivery solutions for IoNT.

## 1.1 Research Challenges

Data delivery for IoNT confronts challenges attributed to the characteristics of both EM-WNSNs and existing network infrastructures. For the sake of effectiveness and efficiency, the networking and communication schemes for nano-devices should own behaviour properties that consider the impacting factors discussed below.

1. **Dynamic THz channel conditions.** The THz band is the next frontier of wireless communications [27]. It offers wide bandwidth capable of supporting high speed communications with low interference to existing frequency bands. However, the signal transmission is affected by strong molecular absorption. On the one hand, molecular absorption makes the channel frequency-selective [14]. This is referred to as the frequency-selective feature in the THz band. On the other hand, the channel condition becomes dynamic in scenarios where the composition of molecules changes over time (e.g. intra-body sensing and out-door monitoring). From the perspective of networking, this characteristic leads to dynamic topology.
2. **Low capacity of nano-devices.** The second challenge is the resource constraints of nano-devices. The small size of nano-devices limits the energy, storage, and computation capacity. Therefore networked nano-devices are expected to execute simple tasks with low resource requirements. Consequently, the communication mechanisms of EM-WNSNs should be designed towards simplification and low costs.

3. **Limited and dynamic IoT bandwidth.** For EM-WNSNs connected to the Internet, the sensed data of nanosensors are first aggregated at nano-sinks and then forwarded to the gateway. Next, the gateway transmits the data to the access and backhaul networks via IoT access and backhaul links. In this thesis, the bandwidth and links provided by the IoT access and backhaul network for EM-WNSNs are collectively referred to as the IoT bandwidth and IoT links. Unfortunately, the current access and backhaul technologies for Machine-Type Communications (MTC) such as LoRa [28], LTE-MTC [29], and NB-IoT [30] are designed for low-bandwidth access; thus, they cannot accommodate the high throughput of EM-WNSNs especially with bursty transmissions [31].

When the mismatch between EM-WNSNs throughput and IoT bandwidth happens, the resource utilization efficiency of both nanonetworks and access networks deteriorates. On the one hand, for the EM-WNSNs throughput that exceed the bandwidth, traffic policing mechanisms at the gateways drop data or traffic shaping mechanisms buffer data and this leads to either unnecessary energy consumption for nano-devices or the demand for large buffer at IoT gateways. On the other hand, reducing EM-WNSNs throughput not only affects information quality [32] but also underutilizes the allocated bandwidth.

To achieve high resource utilization efficiency of the overall network, the throughput of nanonetworks should match the IoT bandwidth.

## 1.2 Contributions

This thesis contributes to the area of EM-WNSNs in the context of IoNT. Specifically, lightweight cross-layer data acquisition and dissemination solutions are designed and evaluated for nanonetworks to connect nano-

devices to the Internet under constrained channel conditions and IoT bandwidth. The layer 2 and layer 3 networking solutions implemented in this thesis, which target high resource utilization efficiency, further realize the functionality of nanonetworks in the current and near-future IoT architecture. The detailed contributions are listed below and summarised in Table 1.1:

Table 1.1: Contribution summary

Contribution	Application scenario	Message unit	Function
CForward	Multi-hop Any-scale	Any	Layer 3
OP Polling	Multi-hop Any-scale	Packet	Layer 2
TEForward	Multi-hop Any-scale	Packet	Layer 3
EAPIS	Single-hop Small-scale	Pulse	Layer 2

- High Speed Forwarding for THz networks.** The behaviours of forwarding schemes in the multi-hop THz networks is studied. A Channel-aware Forwarding (CForward) scheme, which takes into consideration the particularities of THz band, is developed to improve the end-to-end hop channel capacity of THz networks. Performance evaluation is conducted on end-to-end capacity and delay with various network sizes and molecular absorption levels. In comparison with classical geographical forwarding schemes, the channel-aware forwarding improves the channel utilization efficiency of THz networks. Although it demands computational capacity, this work is the first forwarding solution that takes into consideration the end-to-end capacity of THz networks.
- Data Acquisition for query-based IoNT.** The On-demand Probabilistic polling (OP polling) is developed to efficiently extract data from EM-WNSNs under dynamic channel conditions and IoT bandwidth. Under those constraints, resource utilization efficiency is lim-

ited since both EM-WNSNs throughput and IoT bandwidth are dynamic. OP polling matches the bandwidth demand of sensor networks and the actual bandwidth allocated for IoT devices for optimal bandwidth and energy efficiency. Through comprehensive performance evaluation, OP polling achieves high resource efficiency and outperform benchmarks.

- **Data Dissemination for Polling-based IoNT.** The TTL-based Efficient Forwarding (TEForward) is implemented for polling-based IoNT under dynamic channel conditions for beacon and message dissemination. By re-purposing polling beacons, the low-complexity algorithm achieves high energy efficiency close to the optimum in a dynamic topology. TEForward eliminates the requirement of complex hardware for channel condition assessment that is beyond the capacity of nano-devices, which fits the characteristics of EM-WNSNs.
- **Data Acquisition for Event-based IoNT.** The Enhanced Adaptive Pulse Interval Scheduling (EAPIS) is a pulse-level access solution developed for small-scale event-driven IoNT to match EM-WNSNs throughput and limited IoT bandwidth. It achieves this goal by scheduling THz pulses following the ideal pulse arrival pattern that is determined by the IoT access bandwidth allocated for nanonetworks. Thus, on event occurrence, pulses from nanonetworks will arrive at the IoT gateway without triggering traffic policing that drop pulses, which indicates high bandwidth and energy efficiency.

Overall, this thesis contributes novel layer-2 and layer-3 networking solutions to EM-WNSNs with new constraints considered. Compared with existing solutions, the contributed networking approaches utilize lightweight operations to address multiple IoNT constraints that include time-varying THz channel conditions and dynamic IoNT bandwidth at the same time. The contributions of this thesis offer feasible and implementable ways to bridge EM-WNSNs to the context of IoT.

## 1.3 Organisation of Thesis

### Organisation Overview

This thesis is organized as follows:

- **Chapter 2** presents a comprehensive review of background and related work in the field of IoNT. First, the overview of IoNT is discussed in the aspects of the development of nano technologies, the latest status of nano networking, the characteristics of THz band, and the potential nano-scale applications to be enabled. Next, Medium Access Control (MAC) solutions and network layer techniques for IoNT are reviewed. Eventually, considering the limited number of literatures in the field of EM-WNSNs, the polling-based data collection in WSNs is also discussed.
- **Chapter 3** presents a preliminary study on multi-hop forwarding for THz networks. CForward is designed and evaluated. CForward provides an initial trial for exploring the effect of multi-hop forwarding on end-to-end THz channel capacity.
- **Chapter 4** describes the data acquisition for query-based IoNT. First, the existing polling schemes designed for WNSNs are reviewed. Next, the bandwidth-aware OP polling for nanonetworks is designed, evaluated, and analysed.
- **Chapter 5** introduces the lightweight TEFoward for polling-based IoNT under dynamic channel conditions. TEFoward is evaluated against the forwarding with optimal energy efficiency and other benchmarks.
- **Chapter 6** presents the details of the pulse arrival scheduling solution namely EAPIS in the context of IoNT. First, the ideal pulse arrival pattern, which optimizes both IoT bandwidth efficiency and

EM-WNSNs energy efficiency, is characterised. Then, EAPIS is evaluated and discussed. Finally, mathematical models for key performance metrics are developed for guiding network planning.

- **Chapter 7** concludes the content and outcome of this thesis. Several potential research paths are derived for the development of IoNT.

### **Organisation of Technical Chapters**

This thesis includes four technical chapters, which consists of Chapters 3, 4, 5, and 6, to present the details of research contributions. Those technical chapters are organized following two potential logics.

The first logic is the network coordination of proposed approaches. Technical chapters are presented from centralized networking solutions to distributed networking solutions. Chapters 3, 4, and 5 demonstrate centralized methods that address layer-2 and layer-3 networking problems whereas Chapter 6 focuses on distributed layer-2 algorithm. Specifically, Chapters 4 and 5 are stacked as a polling system covering both data collection and dissemination.

The second logic is the message unit adopted by the proposed solutions, which transits from packet-level solutions to pulse-level solutions. Chapters 3, 4, and 5 require packets whereas Chapter 6 is a pulse-level mechanism.

# Chapter 2

## Background and Related Work

In this chapter the background of IoNT is reviewed in the following aspects: the development of graphene-based nano technologies, THz channel, potential applications, MAC layer, and network layer of EM-WNSNs.

### 2.1 IoNT: Overview

Thanks to the development of graphene-based nano technologies, the sensing resolution of WSNs is now downscaled by nanosensors to a molecular level. Taking the advantage of properties of nano materials, nanosensors with nano-meter scales could sense nanoscale events however the individual nanosensor shows extremely limited capacity due to its tiny size. Therefore, they are expected to form nanonetworks for complex sensing tasks. Nanonetworks are further developed to IoNT by establishing communication links towards network infrastructures that are connected to the Internet. There are mainly two kinds of nanonetworks classified according to the communication medium adopted: molecular-based nanonetworks and EM-based Wireless NanoSensor Networks (EM-WNSNs), as shown in Fig. 2.1.

Extensive research on molecular-based nanonetworks has been conducted in recent years [33–46]. Molecular-based nanonetworks are com-

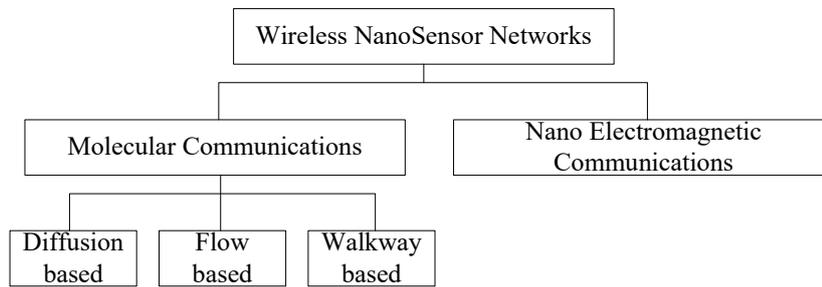


Figure 2.1: Communication options for wireless nanosensor networks [6]

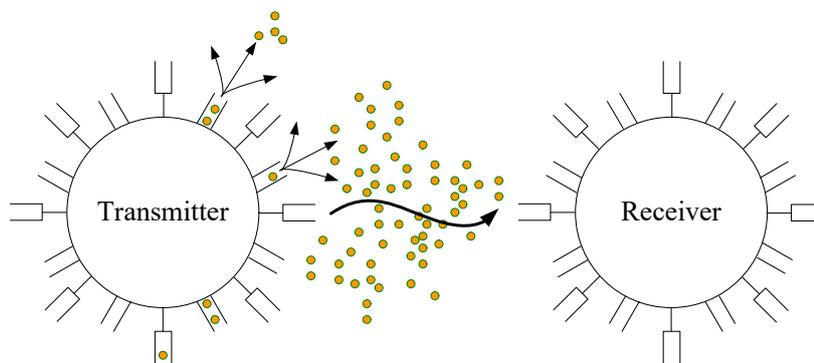


Figure 2.2: Molecular communications [7]

posed of densely deployed nano machines such as chemical sensors, nano valves, molecular elevators, and nano switches. Nano machines include both artificially engineered devices or natural organisms such as cells from biological systems. As depicted in Fig. 2.2, communications among nano machines are carried by molecules with various means of encoding and dissemination. Specifically, messages are encoded using either the concentration of molecules or the structural information of biological particles. Encoded molecules are transported from the source to the destination via three methods: molecule-enabled walkway, guided flows, and random diffusion. Molecular-based nanonetwork could employ existing biological structures as nano machines via function mapping, however, the application of such networks is limited due to the following reasons: first, the data rate is low due to the slow propagation of molecules in medium like

liquid and air; second, the communication link is vulnerable to environmental factors such as temperature; third, releasing molecules in vivo risks introducing interference to the functionality of organs [47].

In comparison, graphene-based nano-devices in EM-WNSNs utilize electromagnetic wave in the THz band as the means for communication [6]. The high sensitivity of graphene and the wide bandwidth of THz band could enable a wider range of applications. In the rest of this section, the detailed context of EM-WNSNs is discussed in four aspects: the characteristics of graphene, peculiarities of THz communications, network architecture, and potential applications.

### 2.1.1 Graphene-based Nano Technology

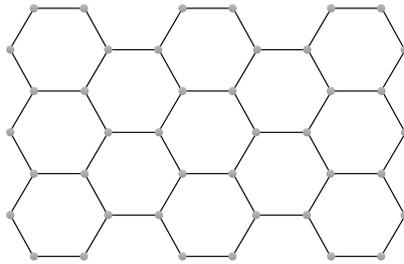


Figure 2.3: Structure of graphene [8]

Since graphene was discovered, it quickly developed into one of the actively researched topics that attracted the world's attention [8, 48]. As shown in Fig. 2.3, graphene is composed of one single layer of carbon atoms tightly packed into a two-dimensional honeycomb lattice. It has several significant properties that make it outstanding in comparison with other materials:

- High strength: it is the strongest material that has ever been measured [49].
- High fracture toughness: the maintainability of graphene is impressive while facing the force of impact [50].

- High electron mobility: the electron mobility is high for a wide range of temperature [8].
- High thermal conductivity: graphene demonstrates extraordinary thermal conductivity under certain conditions in comparison with other materials [51].

Based on the properties of graphene, a wide range of potential applications could be supported. For instance, tissue engineering [52,53], drug delivery [54], body armour manufacturing [55], and biomolecule detection [56].

Amongst all, the graphene-based integrated nanosensors, which are composed of sensors [57], antennas [58], transistors [59] and capacitors [60], are envisaged to be one of the most valuable outcomes. Nowadays, Graphene NanoRibbons (GNRs) and Carbon NanoTubes (CNTs) are being developed to manufacture components of the integrated nanosensor such as THz signal modulators [61], THz plasmonic antennas [11, 58, 62–65], sensing units [57,66,67] and transceivers [68].

However, it must be realized that in comparison with the mature status of macro-scale manufacturing and modelling, there is still a gap between the state of art of graphene-based nanotechnology and their wide-scale deployment. Therefore, graphene is expected to be embedded with the traditional materials such as SiGe and GaN in a hybrid way [27].

### 2.1.2 THz Channel

Because of the small antenna size of nano-devices, the THz band becomes the potential communication frequency for EM-WNSNs [27]. In this section, the peculiarities, challenges, channel modelling and modulation of the THz band are reviewed.

### Peculiarities of the THz Band

The THz band (0.1 THz – 10 THz) is regarded to be the next frontier of wireless communications [27]. In comparison with the adjacent frequency bands, THz is the only one that provides communication links with an information rate up to the level of Tbps. For the frequency band below THz, the channel capacity is limited by the bandwidth. For the frequency band above THz, signal propagation confronts the regulated energy level and the high path loss. Consequently, the information rate is reduced [27]. Nowadays the THz band has become a research focus because

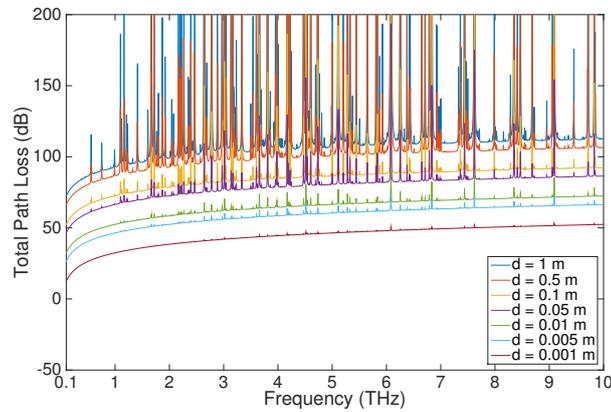


Figure 2.4: The path loss in the THz band (USA Model, Mean Latitude, Summer, H=0 [9])

it could benefit a large number of applications ranging from macroscale to nanoscale [27,69,70]:

- At the macroscale, the sufficient bandwidth resource of THz band could enable short-range high-speed directional wireless links. Therefore, it becomes the best technique for small cells and their access and backhaul networks in 5G [71], wireless local/personal area networks [72,73], and military wireless networks [74].
- At the nanoscale, the small size of nano antennas naturally makes the THz band as one of the operational frequency bands. The poten-

tial nano-scale applications of THz band include in-vivo monitoring applications [75], chemical defences [76], and wireless on-chip communications [77].

The standardization of near-THz band has started in the year of 2014 by IEEE 802.15 WPAN Task Group 3d 100 Gbit/s Wireless (TG 3d (100G)) whose goal is to standardize the THz band to support high-data-rate links [78].

### **Networking Challenges of the THz Band**

Despite the wide bandwidth, the defects of THz band make the potential utilization challenging. The emerging challenges from using the THz band affect every protocol layer of networks:

- At the physical layer, first, generating, detecting, and synchronizing THz signals demands sophisticated design and manufacture for hardware [79]. Besides, due to molecular absorption caused by molecular resonance, the path loss and noise temperature in the THz band are frequency-selective (as shown in Fig. 2.4) [14]. This is detrimental to the available bandwidth in long-distance transmission especially for certain kinds of molecules such as water. Meanwhile, the transmission distance is limited due to the high frequency [80]. These deficiencies need to be addressed to efficiently use the THz band.
- At the link layer, MAC solutions and error coding schemes need to capture the new features of THz links since interference and collision are no longer influential due to the short transmission time that benefits from high data rates [15].
- At the network layer, the first challenge is the end-to-end channel utilization with distance-related bandwidth. Forwarding and routing process must take into consideration the environmental humid-

ity that impacts the path capacity. Second, appropriate node addressing is needed for the small-scale high-speed THz networks [10].

### THz Band Channel Modelling

In the THz band, the total path loss  $A_T$  is the product of the spread path loss  $A_S$  and the molecular absorption loss  $A_{abs}$  [14]. The path loss is a function of the transmission distance  $d$  and the frequency  $f$  and is expressed as:

$$A_T(f, d) = A_S(f, d) \cdot A_{abs}(f, d) = \left( \frac{4\pi df}{c} \right)^2 \cdot e^{k(f)d} \quad (2.1)$$

where  $c$  is the speed of light  $3 \cdot 10^8$  m/s, and  $k(f)$  is the molecular absorption coefficient obtained from the HITRAN (High resolution TRANsmission) molecular absorption data-base [9]. The total path loss  $A_T$  as a function of THz frequency are shown in Fig. 2.4. From Fig. 2.4 it can be seen that the frequency selective feature is dominant for transmission involving longer distances  $d$ .

The noise in the THz band is mainly due to molecular absorption noise  $N_{abs}$ . The power spectral density (P.S.D.) of  $N_{abs}$  is given by  $N_{abs}(f, d) = k_B T_0 (1 - e^{-k(f)d})$ , where  $k_B$  is the Boltzmann constant and  $T_0$  is the reference temperature 296K.

The THz channel capacity is expressed as a summation of the capacity of each narrow sub-channel, and the capacity of each narrow sub-channel is given by Shannon's channel capacity [14]. This is achieved by dividing the THz band into  $n$  sub-channels, and the THz channel capacity is given by:

$$C_{THz}(d) = \sum_{i=1}^n B \log_2(1 + SNR(f_i, d)) \quad (2.2)$$

$$SNR(f, d) = \frac{S(f)}{A_T(f, d)N_{abs}(f, d)} \quad (2.3)$$

where  $n$  is the number of sub-channels,  $f_i$  is the centre frequency of the  $i$ -th sub-channel,  $S(f)$  is the P.S.D. of the transmitted signal, and  $B$  is the

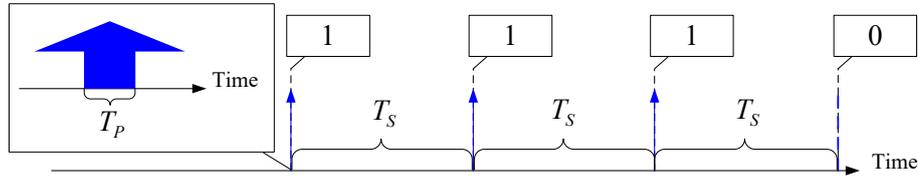


Figure 2.5: TS-OOK modulation

bandwidth of each sub-channel.

In conclusion, THz band has a very large bandwidth to enable high data rates for IoNT. To efficiently utilize the potential channel resource, communication and networking solutions that are aware of the channel peculiarities are needed.

### THz Pulse Modulation

To accommodate the limited energy capacity and small antenna size of nano-devices, an asymmetric On-Off Keying modulation Spread in Time (TS-OOK) has been proposed [15]. As presented in Fig. 2.5, in TS-OOK, bits are modulated by the presence and absence of short pulses. Specifically, each bit “1” is modulated via the presence of one short pulse and each bit “0” is modulated by silence. In TS-OOK, the pulse duration  $T_P$  is as short as  $100\text{ fs}$  and the time interval between two pulses  $T_S$  is far longer than  $T_P$ . Because of the mechanism of TS-OOK, the communication systems benefit in the following aspects:

- This modulation scheme makes the bit sequence of a single user mostly filled with “silence” therefore effectively alleviating the burden of Medium Access Control (MAC) complexity. Collision avoidance is no longer a major concern due to the low collision probability, which implies improved system reliability. However, solutions are needed for maximizing medium utilization efficiency.
- By adopting coding schemes which minimizes the ratio of the num-

ber of bit “1” to the number of bit “0” in packets, TS-OOK could achieve low energy consumption so as to increase the network life time.

- The long time interval between two pulses could enable multiple users to transmit concurrently. The interleaved transmission is not impacted by collision and interference of pulse width is short. This benefits the aggregated throughput of EM-WNSNs.

However, due to the short pulse duration, communication and networking functions such as signal detection and synchronization become challenging for nano-devices composed of basic components that only support simple operations.

### 2.1.3 Nano-device and Network Architecture

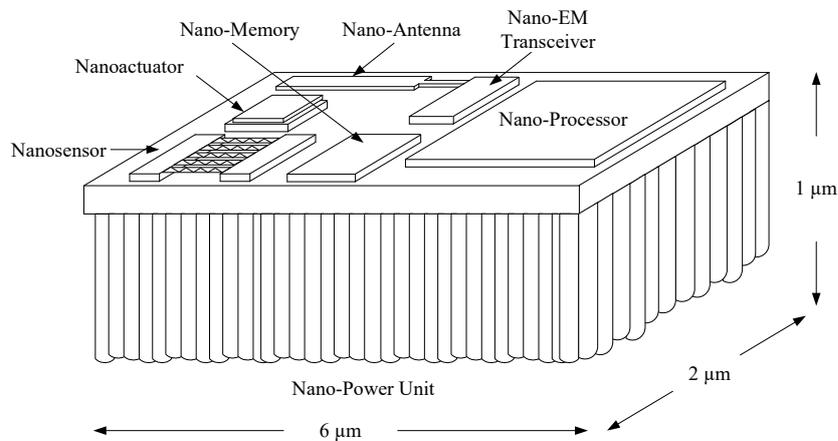


Figure 2.6: An integrated nano-device [6]

In EM-WNSNs, each integrated nano-device is envisaged to have a size of around  $10\text{-}100 \mu\text{m}^2$  with basic capacity for simple tasks. As presented in Fig. 2.6, an integrated nano-device is comprised of the following blocks [6]:

- Sensing unit: different measured magnitudes demand different nanosensors such as physical sensors for measuring mass, force, or pressure, chemical sensors that measure concentration or types of molecules, biological sensors for monitoring bio-molecular process such as DNA interactions [81].
- Actuation unit: actuation units enable the interaction between nano-devices and the physical environment. Based on the types of interacted objects, actuation units are classified into two categories that are physical nanoactuators and bio-chemical nanoactuators [82].
- Power unit: to date, two kinds of energy sources are studied to power nano-devices. The first one is lithium nano batteries [83] that are manufactured using ZnO. Alternatively, energy harvesting techniques that are more flexible and reliable are adopted [84].
- Processing unit: nano processors could be based on two types of materials: graphene and silicon. Graphene-based processing units [85] have small size and potentially high computing speed that benefit from the physical properties such as the one-dimension structure and ballistic transport of electrons. In comparison, silicon-based units [86] have larger size and slower speed.
- Storage unit: generally, nano memories represent binary bits by utilizing the presence and absence of one atom. So far, it is reported that silicon [87] or magnetic atoms [88] are studied for manufacturing nano storage units.
- Communication unit: nano communication units could be realized in two ways. The first option is composed of nano antennas and nano-EM transceivers [63, 89]. Alternatively, a communication unit could also be enabled via a single CNT resonator [90].

Individual nano-devices have limited physical capacity, therefore, they must be coordinated to conduct complex tasks. The overall network archi-

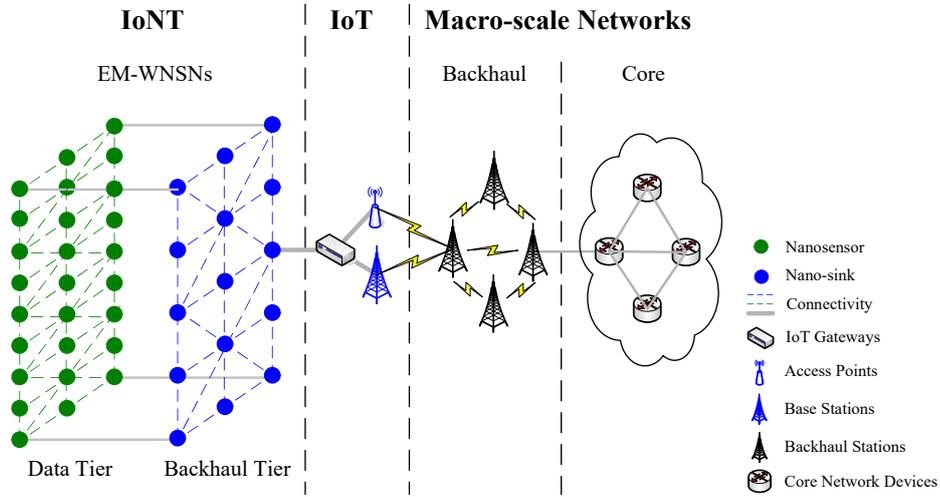


Figure 2.7: Overall network architecture of IoNT

architecture of IoNT is depicted in Fig. 2.7. Nano-devices are coordinated and aggregated by the IoT gateway that bridges EM-WNSNs and the macro-scale infrastructures such as access and backhaul networks and core networks. In detail, network devices are classified as follows [10,91]:

- **IoNT domain**: nanosensors with small size and low capacity form the data tier that is densely deployed in a non-invasive way. The data tier aims to sense events and generate data for the backhaul tier composed of nano-sinks that are designed for data aggregation. Nano-sinks have lower density but higher physical capacity than nanosensors so they are responsible for relatively complex activities.
- **IoT domain**: IoT gateways establish the communication link between nano-devices and Internet infrastructures. To do so, the function of converting signalling and messages between the nano scale and the macro scale is required. This function is carried by either the gateway [22] or an extra nano-macro interface [10].
- **macro-scale networks**: traditional network infrastructures such as access network, backhaul networks, and core networks are deployed

to connect the remote users such as the IoT cloud [92].

The network architecture introduced above provides a guideline for IoNT design and deployment. Depending on the specific applications, changes may apply to optimize network performance.

#### **2.1.4 IoT Access and Backhaul Solutions**

The IoT access and backhaul networks are responsible for transferring M2M data from machine networks to the Internet. The access and backhaul networks for IoT are supported by a wide range of communication techniques with different characteristics [93]. Existing IoT access and backhaul solutions are classified by coverage and discussed below.

Wifi-based technologies offer short-range access and backhaul services to IoT devices. IEEE 802.11ah [94] with low infrastructure cost is a novel protocol designed for IoT. Compared with other protocols from IEEE 802.11 family, IEEE 802.11ah adopts an operation frequency below 1 GHz therefore it has longer coverage up to 1 km. Besides, IEEE 802.11ah supports up to 8100 IoT stations in a single cell.

In comparison with wifi-based solutions, cellular-based technologies such as LTE [95], LTE-A [96], and 5G [97] offer ubiquitous access and larger coverage because of the widely deployed cellular stations. However, the potential energy cost of accessing cellular stations is high.

In order to achieve better coverage and energy efficiency, LPWAN [98] is considered as the access and backhaul solution for IoT. Adopting narrow band in low frequency, LPWAN provides coverage up to 40 km and battery life up to several years. Although the capacity of LPWAN is low, it fits the traffic pattern of existing and near-future IoT applications such as the recently proposed LPWA-based air quality monitoring system [99] which utilizes LPWAN to transfer data from sensors to the latest IoT cloud [92].

For IoNT with high throughput, current IoT access and backhaul solutions are potentially challenged in terms of the available bandwidth.

The access and backhaul bandwidth allocated to IoT applications is limited and dynamic because of the following reasons: 1) the low priority of M2M traffic for access and backhaul solutions where M2M and H2H communications coexist; and 2) the total access and backhaul bandwidth is impacted by weather conditions if microwave links are adopted to form access and backhaul networks [100]. As a consequence, mechanisms designed for connecting EM-WNSNs with Internet should be aware of the bandwidth limit for efficient utilization of IoT access and backhaul networks.

### **2.1.5 IoNT Applications**

Benefiting from the merits of graphene-based nano technologies and THz communications, the cutting-edged IoNT is enabling evolutionary sensing applications with unprecedented resolution. With this advantage, deep insights could be explored by the remote applications that conduct data analysis. Based on the purpose and scenario, IoNT applications are classified into four categories that are biomedical applications, environmental applications, industrial applications, and military applications [6].

- **Biomedical applications:** biomedical application is one of most promising scenario that is directly enabled by IoNT. Unlike traditional on-body monitoring based on macro-scale WSNs, nano-scale intra-body sensing provides molecular-level detecting accuracy which achieves real-time monitoring. There are mainly two types of biomedical applications based on the functionality of nano-devices: health monitoring and drug delivery. Nanosensors could monitor multiple biological agents in blood [101]. Recently, several solutions for detecting in-vivo objects such as blood [102], lungs [103], and hearts [104] were proposed, which presents an example of IoNT-enabled health monitoring. Drug delivery [105] is another application that can only be implemented by IoNT. Multiple nano-devices could be coordinated

to release a dynamic amount of drugs for stabilizing in-vivo environment.

- **Environmental applications:** the messages that control the natural behavioural processes of plants are carried by the chemical composites that are released in the air. Therefore, nanosensors with high bio-chemical sensitivity can be implanted into plants for monitoring and actuating operations. The current IoNT environmental applications include plant monitoring and plague defeating. Plant monitoring relies on the bio-chemical sensors in nano-devices. Upon sensing the target chemical compounds, nano-devices will be coordinated for data aggregation and dissemination. For efficient communications, several research outcomes have been reported such as channel modelling and measurement [106–108], performance analysis [109], and dynamic channel allocation [23, 110]. In comparison, plague defeating applications utilize nano-devices to pro-actively release corresponding chemical particles for adjusting the behaviours of plants [111].
- **Industrial and consumer applications:** deploying IoNT in industrial and consumer scenarios could largely enhance the manufacture and innovation of products. Several application examples could be observed: ultra-high sensitivity touch surfaces could be realized by coordinating densely-deployed touch sensors [112]; real-time chemical reaction monitoring [113, 114] is enabled by chemical sensors that are embedded in the catalyst surfaces; Wireless Network on Chip (WNoC) [77] achieves fast inter-core transmission because of the broadcast nature and high speed of THz communications; Software Defined Materials (SDM) [115] allows materials to have run-time configured properties for versatile functions.
- **Military and defence applications:** IoNT is envisaged to benefit two types of military applications: the nuclear and bio-chemical detec-

tion, and physical damage detection. The former one could be implemented by using the chemical reaction monitoring system applied for industrial applications, however, with higher demands for precision and speed. In physical damage detection, nano-devices are densely deployed to cover objects such as textiles, structures, and vehicles for molecular-level detection [116].

## 2.2 IoNT: MAC Layer

In wireless networks, MAC schemes play a critical role in arbitrating access to the shared wireless channel. This task is accomplished in two dimensions that are the frequency domain and time domain which allocate channel frequency and time slots of the medium, respectively.

MAC schemes for EM-WNSNs differ from the traditional ones in two aspects. First, in frequency domain, Channel Allocation (CA) needs to consider not only the co-channel interference but also the frequency-selective channel response of THz band. Many IoNT applications that exhibit periodic humidity change would confront the frequency-selective channel condition of THz band. In those applications, THz channel capacity fluctuates due to the biological processes that change the moisture ratio of channels. This characteristic increases the difficulty of algorithm design and hinders the direct application of corresponding solutions in the field of traditional WSNs. Second, in the time domain, collision avoidance and interference mitigation are no longer the first target because of the short pulses and long inter-pulse distance adopted by TS-OOK. In this case, increasing resource utilization efficiency under time-varying channel conditions becomes significant.

In the rest of this section, the existing MAC solutions in both the frequency domain and time domain are discussed.

### 2.2.1 Frequency Allocation

To optimize system performance such as overall channel capacity and successful transmission probability, the Genetic Algorithm (GA)-based frequency selection schemes [23] were proposed as the first CA solution for this research problem. In this work, the optimization problem is formulated as a double-objective problem which attempts to maximize both cluster-level fairness and overall channel capacity. Based on the status of frequency overlap in the channel capacity objective, three frequency selection methods, which are the normal optimization, constraint-based optimization and swarm-Based optimization, were proposed. Next, the GA-based solution, which adopts the weighted sum of double objectives and defines each chromosome as a pair of cluster and frequency allocated for each nano-device, is designed to solve the proposed optimization problems. Finally, the proposed methods as well as the random benchmark were investigated and evaluated via successful transmission probability, fairness, channel capacity, and power consumption. From performance evaluation, three proposed methods show their own advantages in different performance metrics as a result of diverse design regarding frequency overlap.

The above work is then extended to support nano-scale crop monitoring under varying moisture levels [117]. In this scenario, the moisture ratio of vegetation changes rapidly due to hydration and dehydration, which leads to fluctuating channel capacity over time. Besides the frequency allocation methods in the previous work, the off-line approach and two-phase approach were proposed. In details, the off-line approach retrieves the best transmission frequency from a database that records the frequency allocation for different medium conditions. The two-phase approach takes the off-line approach as the first phase and the swarm-based optimization as the second phase so as to mitigate collision and interference. The final evaluation investigates the performance of frequency selection in dynamic moisture conditions and the difference between static and dynamic configurations of different approaches. In the subsequent work [110], all the

frequency selection schemes were evaluated via a THz propagation model in plants, which shows that the two-phase approach with GA-based optimization achieves comparable throughput in comparison with exhaustive search for various leaf density.

The above-mentioned work aims to implement frequency allocation for channels that are mostly impacted by water. However, for chemical reaction monitoring that demonstrates dramatic and rapid change in channel composition, the MDP-based method provides a more appropriate solution. The MDP-based frequency hopping [26] models the chemical reaction process as a discrete-time state transition and calculates the optimal transmission frequency at each time slot. Because of the SNR-based rewards, the MDP-based hopping achieves the best SNR, channel capacity, and BER performance in comparison with the best channel method, two off-line methods, and the random hopping. However, implementing MDP-based frequency hopping requires the global knowledge of the up-to-date channel composition. Therefore, off-line methods with sub-optimal performance are appropriate for nano-scale devices.

### 2.2.2 Time Allocation

Based on the coordination mechanisms, time-allocation MAC schemes for EM-WNSNs are classified into two categories: the centralized MAC that demands central time allocation for individual nodes and distributed MAC which relies on autonomous operations of individual nodes.

#### Centralized MAC Protocols

To achieve perpetual operations for energy-harvesting EM-WNSNs, the energy and spectrum-aware MAC protocol [18] is designed. This protocol consists of two major techniques that are Critical Transmission Ratio (CTR) and throughput-and-lifetime optimal scheduling. Firstly, CTR calculates the maximum ratio between transmission time and energy harvesting time

which ensures infinite network lifetime. Next, based on CTR, the optimal scheduling is conducted by the controller to allocate individual nanosensors with time slots that achieve the optimal overall throughput and network lifetime with no collisions. In this way, several nanosensors could transmit in parallel because of the interleaved transmission. Regardless of the high data rate and infinite network lifetime, this protocol only supports single-hop networks wherein nanosensors are directly connected to the controller, which limits the applicability for multi-hop networks.

Considering the limited capacity of nano-devices, receiver-initiated MAC schemes become potential solutions for EM-WNSNs because of the fine-grained coordination which benefits communication efficiency. The first MAC solution that follows this design mode is the Receiver-Initiated Harvesting-aware MAC (RIH-MAC) [19] proposed for single-hop networks. In the centralized RIH-MAC, the controller broadcasts beacon packets for all nodes when it is ready to receive data for synchronization and data acquisition. Afterwards, each node decides whether to transmit based on a participation probability that is related to the local energy status of nodes. This probability aims to allow only one node to transmit. Finally, the controller sends an acknowledgement upon the successful reception of data. The centralized RIH-MAC exhibits high probability of successful packet reception with well-constrained collision probability.

Another approach proposed for energy-harvesting EM-WNSNs is the energy-harvesting aware MAC protocol [21] proposed for Body Area Nano NETWORKS (BANNET). In this work, nodes are clustered and coordinated by the cluster head for medium access and routing. This MAC protocol reduces collisions and improves network lifetime by deactivating the physical reception function of nodes when they are not supposed to receive any message during the handshake process with the controller.

A timing channel-based MAC protocol called Timing Channel for Nanonetworks (TCN) MAC [20] is applied in low-rate EM-WNSNs. Timing channels encode messages in the silence between subsequent events. The pro-

posed TCN MAC is a centralized scheme that all the nodes are synchronized and managed by a gateway responsible for central control operations. Briefly, TCN is composed of two phases. In phase one, the gateway allocates the transmission time in each frame for packets of each node in an overlap-free pattern. In the second phase, packets are transmitted at scheduled time and recovered by the gateway based on the timing information of individual nodes. Compared with traditional access control methods, TCN shows advantages in terms of successful transmission rate and network lifetime.

Most of the time allocation methods only consider static channel environment. However, similar to frequency allocation, time allocation for EM-WNSNs also need to address the challenge of dynamic channel conditions. To accommodate the periodically-varying channel condition caused by respiration cycles in human lung monitoring, a sweet spot estimation (SSE) [103] was proposed. In SSE, the nano collector estimates SNR of the periodic beacons sent by nano-sinks so as to forecast the next time window with low channel attenuation that is suitable for low-power transmission. Although no specific MAC operations are developed, SSE implements the first step that determines the appropriate time to access channel. The only limitation is the computational capacity and time precision of nano collectors demanded by SNR calculation.

### **Distributed MAC Protocols**

Because of TS-OOK, collision avoidance is no longer the major challenge for EM-WNSNs. However, consecutive collisions may happen once the first pulses of two nodes collide. To solve this problem, the first MAC protocol for EM-WNSNs, namely the Physical Layer Aware Mac protocol for Electromagnetic nanonetworks (PHLAME) [17, 118], was proposed. In PHLAME, an enhanced version of TS-OOK modulation called Rate Division Time Spread On-Off Keying (RD TS-OOK), in which different users randomly chose pairwise coprime rate codes, is adopted. Because of the

heterogeneous pulse intervals, consecutive collisions after the initial one are unlikely to happen. Afterwards, a two-way handshake is initiated by the sender which notifies the receiver its data rate. Next, the receiver decides whether to accept the transmission and dynamically specifies the coding weight and repetition code for each transmission based on physical layer demands. Based on the RD TS-OOK and the handshake process, PHLAME achieves desirable performance in terms of energy efficiency, latency and throughput.

The random process for pulse rate selection of PHLAME poses a problem for multiple nodes that choose the same rate and collide for successive pulses. To solve this problem, the Enhanced Rate Division (ERD) MAC [119] was proposed. The core of this work is the prime mod algorithm which generates unique rates for individual nodes so as to implement RD multiple access with minimal collision and low complexity.

To make the controlling mechanisms simple, two lightweight MAC schemes [22] were proposed for EM-WNSNs: the transparent MAC and the smart MAC. In the transparent MAC, the packets are transmitted from the network layer to the physical layer without any executing any control in the MAC layer. In smart MAC, before data transmission, handshakes are initiated by senders for neighbour discovery. Afterwards, a tunable random backoff time is arranged to avoid collisions. However, those two lightweight solutions lack the support for key characteristics of EM-WNSNs such as pulse interleaving and energy-harvesting EM-WNSNs.

The PHLAME, ERD MAC, transparent MAC, and smart MAC are all sender-initiated methods which may fail to optimize system performance. For example, in PHLAME, senders may face the chance of transmission rejection due to receiver-side saturation that introduces energy consumption and latency. To improve the network performance for energy-harvesting nanonetworks, the distributed version of the previously discussed RIH-MAC, which is called DRIH-MAC [19, 120], was proposed. In DRIH-MAC, the medium allocation is obtained from an edge-colouring algo-

rithm which ensures different neighbours of a node transmit in different time slots. Besides, an energy-aware MDP is applied for nodes to predict the energy status of neighbours for successful communications. DRIH-MAC shows a high data-reception percentage close to the ideal case without introducing high energy consumption due to control messages.

Receiver-initiated MAC protocol could incorporate sliding window mechanisms for flow control as well [121]. For wireless nodes communicating with each other, latency happens due to reasons such as receiver unavailability and long propagation distance. To reduce the delay caused by receiver unavailability, for each handshake initiated, the receiver specifies the number of data packets it is willing to receive considering its current queueing status and energy level. Next, the sender only sends the on-demand data volume as response. In this way, both throughput and packet delivery ratio are enhanced.

Nanosensors face challenges from many aspects such as energy limitation and storage limitation. However, most of the MAC solutions proposed lack the joint consideration of both energy storage and memory space. To achieve optimal timely throughput with perpetual operations of nano-devices for delay-constrained applications, two optimal MAC scheduling algorithms which are Distributed Maximum Debt Scheduling (DMDS) and DMDS- $\alpha$  were proposed [122]. The behaviours of nodes in the proposed work are determined by two parameters. First, for both DMDS and DMDS- $\alpha$ , the minimal time slot that achieves perpetual operations is decided considering the energy harvesting process. Next, the channel sensing periods for DMDS and DMDS- $\alpha$  are determined based on the queueing status so as to ensure empty queues for the next time slot. In comparison with DMDS, DMDS- $\alpha$  additionally supports different priority for users. Given the minimal time slot and the sensing period, nodes access the channel in a lightweight and distributed manner with endless network lifetime and satisfied packet delivery deadline.

### 2.2.3 Summary of MAC Layer in IoNT

The frequency-allocation and time-allocation MAC protocols are summarized in Table 2.1 and Table 2.2, respectively. For time-allocation, in EM-WNSNs, the intrinsic property of TS-OOK removes the problem of collision but introduces the need for synchronization among nanosensors. Most of the current MAC protocols rely on the synchronized networks which introduces the potential requirement for better centralized control. Therefore, the function of nano-sinks is expected to be explored to a larger extent. Besides, few researchers consider the bandwidth limitation introduced by access and backhaul networks in their scenario.

Table 2.1: Summary of MAC (Frequency Allocation) Protocols in EM-WNSNs

Protocol	Coordination	Channel Condition	Energy Harvesting
GA-based Solution [23,110,117]	Centralized	Dynamic	x
MDP-based Solution [26]	Distributed	Dynamic	x

Table 2.2: Summary of MAC (time allocation) protocols in EM-WNSNs

Protocol	Coordination	Channel Condition	Energy Harvesting	Synchronization	IoT Bandwidth
Energy and Spectrum-aware MAC [18]	Centralized	Static	✓	✓	x
Centralized RIH MAC [19]	Centralized	Static	✓	✓	x
Energy-harvesting Aware Protocol [21]	Centralized	Static	✓	✓	x
TCN MAC [20]	Centralized	Static	x	✓	x
SSE [103]	Centralized	Dynamic	✓	✓	x
PHLAME [17, 118]	Distributed	Static	✓	✓	x
ERD MAC [119]	Distributed	Static	x	x	x
Smart MAC [22]	Distributed	Static	x	x	x
Transparent MAC [22]	Distributed	Static	x	x	x
Distributed RIH MAC [19, 120]	Distributed	Static	✓	✓	x
Receiver-initiated MAC [121]	Distributed	Static	✓	✓	x
DMDS/DMDS- $\alpha$ [122]	Distributed	Static	✓	✓	x

## 2.3 IoNT: Network Layer

The characteristics of EM-WNSNs rule out the use of protocols of traditional WSNs:

- Due to the distance-dependent molecular absorption in the THz channel, the channel condition becomes dynamic. For nano-devices deployed in scenarios with dynamic humidity, the network topology is time-variant.
- Due to the restricted memory, energy and computational capacity, nanosensors are not likely to individually execute complex operations that require a certain level of resources. Instead, network coordination should be delegated to nano-sinks which have better physical capacity than sensors.

Based on the such limitations, from the perspective of nanosensors, the networking schemes of EM-WNSNs should be designed with simplicity. Nano-sinks with better capacity could take centralized tasks for the network coordination.

The existing routing schemes in EM-WNSNs are classified into two categories: centralized routing and distributed routing. In the centralized routing, the routing decision is determined by the nano-sink. In distributed routing, the routing decision is determined by the nanosensors based on their local information.

### 2.3.1 Centralized Routing in IoNT

Routing and forwarding play a significant role in data delivery. Although there are numerous schemes designed for WSNs that could potentially be used for EM-WNSNs, the validation needs to be done taking into consideration the characteristics of EM-WNSNs.

The most typical centralized routing scheme is the routing framework [123] for energy-harvesting EM-WNSNs. On top of the energy and spectrum-aware MAC protocol [18], the proposed routing solution aims to pursue a balanced energy process and optimal throughput. The proposed routing framework defines a hierarchical cluster-based architecture where the nanosensors in one cluster are coordinated by a nano-controller with high resource capacity. The routing decision algorithm is as follows. First, the nanosensor that intends to transmit sends a request to the nano-controller. After receiving the request packet, based on the distance between the nanosensor and the nano router, the nano router then computes an energy-saving probability which is the probability that the multi-hop transmission consumes less energy than the single-hop transmission. Next, The nano-controller determines whether the nanosensor should transmit via single hop or multi hops based on the criteria of energy efficiency. If single-hop transmission is adopted, the nanosensor will transmit to the nano-controller through the allocated time slots. If the nanosensor is instructed to transmit through multiple hops, the nano-controller will compute a Critical Neighborhood Range (CNR) which is the optimal transmission distance to achieve the best tradeoff between throughput and forwarding advancement. A measure called Required Transmission Power (RTP), which indicates the transmission power required to meet the SNR threshold for potential receivers located at a distance equal to CNR, is computed by the nano-controller and allocated to the nanosensor. Finally, the nanosensor tunes its power according to RTP and executes the data transmission. The routing algorithm in the routing framework maintains balanced energy process and high network throughput. As the design of the routing framework fits the characteristics of EM-WNSNs, it offers a promising direction for the future research.

Two other energy-oriented routing protocols [21] were proposed for Body Area Nano NETWORKS (BANNET): the optimal energy-harvesting aware routing protocol and the greedy energy-harvesting aware routing

protocol. In BANNET, the network is composed of nanosensors, nano-routers and a nano-macro interface hierarchically. Data collection is initiated by the interface. Therefore, routing will select the data provider rather than the data forwarder. In the optimal energy-harvesting protocol, the energy efficiency is improved in a cluster level. Precisely, upon receiving the data request from the nano interface, the nano-router will estimate the instant energy levels of nodes collected via the handshaking process to select the node  $i$  which minimizes the cluster-level energy consumption.

In contrast, the greedy energy-harvesting aware routing protocol simply selects the node with the highest energy level in the cluster to be the data provider. The greedy protocol does not have to evaluate the cluster energy level which requires high computational capacity and shows almost the same performances as the optimal protocol. They all outperform the simple flooding in terms of energy efficiency.

### 2.3.2 Distributed Routing in IoNT

Centralized routing shifts the computational load to the central nodes and achieves high effectiveness in the aspect of control however it relies on handshakes that introduce delay and energy consumption. For some application scenarios that demand fast response, distributed methods are more appropriate.

The selective flooding and the random routing [22] were proposed for inner-body health-monitoring systems. In selective flooding, the sending node broadcasts its packet to all its neighbours. The neighbours will forward the packet only when it is not a duplicate to avoid bandwidth consumption. To implement this process, all the nodes store the source ID and packet ID up to 20 packets. Meanwhile, packets can only be forwarded along a positive direction towards higher-level devices (nanosensors  $\rightarrow$  nano-routers  $\rightarrow$  nano interfaces) unless there is no such device being active. Working based on the smart-MAC, the random routing scheme the

sender randomly selects one of its neighbours to be the forwarder. All the nodes store the packet ID and the next-hop ID to avoid routing loops. Additionally, the random selection follows a prioritized order to increase data delivery rate. Both selective flooding and random routing requires memory capacity to implement the processes. Selective flooding outperforms random routing in terms of packet loss ratio because the randomly selected forwarder does not ensure an end-to-end path. However, more physical transmissions happen as a tradeoff.

The Software Defined Material (SDM) is an emerging nano-scale application that favours distributed routing schemes. For this application, the proposed routing solutions are classified into two categories: the anchor-based scheme and non-anchor-based scheme. For anchor-based routing, individual nodes make routing decisions based on their locations inferred from beacon signals sent by anchors. The first anchor-based routing for nanonetworks is the Coordinate and Routing system for Nanonetworks (CORONA) [124]. In CORONA, nodes use the coordinated hop-count information provided by four anchor nodes to infer their geographic locations. After that, packets are routed based on the minimal hop-count principle. The CORONA aims to provide a GPS-free geographic routing solution for 2D networks however it lacks the support for 3D scenarios.

To implement efficient routing for 3D SDM, CORONA is then extended to  $N^3$  [125] routing. The  $N^3$  addresses both node addressing and routing. First, eight anchors are adopted to notify nodes with the hop count from each anchor. Next, nodes select three anchor nodes as the optimal coordinate system that minimizes the number of forwarders for each end-to-end delivery. Finally, individual nodes utilizes trilateration to determine whether they will be the forwarders when a data packet is received. Following those operations,  $N^3$  implements multipath routing between two points in a 3D space with constant complexity.

The operations of both CORONA and  $N^3$  are based on selective flooding which increases connectivity but restrains network efficiency. To re-

duce routing redundancy, the stateless linear-path routing (SLR) [126] was proposed for 3D SDM. SLR is initialized with network addressing via the same mechanism as  $N^3$ . However, in the route decision process, a node acts as a forwarder when it is located in the linear path between the source and the destination. Additionally, the width of the linear path, which controls routing redundancy, is tunable based on the feedback of packet delivery ratio. Further, in the extended work [127], the anchor selection process is optimized to fit the linear path routing.

The above-mentioned anchor-based routing methods require anchors placed in the predefined locations of SDM, which limits the application of such solutions in space with irregular shapes. To route packets for nanonetworks with no coordinate systems, the non-anchor-based routing solutions were proposed. The first non-anchor-based routing is a metaheuristic-based selective flooding (MSF) dissemination scheme for 2D nanonetworks [128]. In this work, the source pro-actively controls the direction of packet propagation by enabling the target gateway while disabling the gateways on other directions.

Next, this selective flooding is simplified to be the lightweight self-tuning data dissemination (LSDD) with low complexity [129]. The proposed work is a receiver-based forwarding scheme. For each transmission, the sender simply broadcasts its packet. The receiver that receives the packet decides whether to join the forwarding process based on a metric calculated from its number of successful receptions.

To further reduce the number of retransmitters, the DEployable ROUting System (DEROUS) [130] was proposed. Unlike other routing solutions for SDM, DEROUS routes packets along radial and circular paths around one beacon point, which is appropriate for nanonetwork with a random or well-defined 2D layout. Briefly, DEROUS contains a deployment phase which does node addressing and a routing phase that routes packets based on the predefined criteria. Compared with other selective flooding schemes, DEROUS successfully restrains the number of retrans-

mitters.

### 2.3.3 Summary of Network Layer in IoNT

Table 2.3: Summary of routing protocols in EM-WNSNs

Protocol	Coordination	Channel Condition	Energy Harvesting	Communication
Energy-saving Routing [123]	Centralized	Static	√	Unicast
Optimal Energy-harvesting Aware Routing [21]	Centralized	Static	x	Unicast
Greedy Energy-harvesting Aware Routing [21]	Centralized	Static	x	Unicast
Selective Flooding [22]	Distributed	Static	x	Broadcast
Random Routing [22]	Distributed	Static	x	Unicast
CORONA [124]	Distributed	Static	x	Multicast
N <sup>3</sup> [125]	Distributed	Static	x	Multicast
SLR [126]	Distributed	Static	x	Multicast
MSF [128]	Distributed	Static	x	Multicast
LSDD [129]	Distributed	Static	x	Multicast
DEROUS [130]	Distributed	Static	x	Multicast

The routing protocols in EM-WNSNs are summarized in Table 2.3. Most

of the routing solutions consider energy efficiency as the key issue and are implemented in a distributed way, which fits the characteristics of EM-WNSNs. However, it can be observed that none of the current work captures the channel peculiarity in the THz band and the limited IoT bandwidth. Therefore, there is a good opportunity for novel research to be done.

## 2.4 Polling in WSNs

Considering the sparsity of research on data acquisition in EM-WNSNs, the data acquisition in WSNs, especially polling, is reviewed in the following. Data acquisition in Wireless Sensor Networks (WSNs) is classified into four modes: event-driven, time-driven, query-based (polling), and hybrid [131].

In event-driven mode, individual sensors generate data and report to the central terminal when pre-defined sensing events are detected. In time-driven data acquisition, sensor nodes periodically generate data following the pre-configured reporting period to obtain global information of the sensing area.

Compared with event-driven mode and time-driven mode, the query-based (polling) approach, whereby sensor nodes commence sensing tasks only when they are queried by the central node, ensures high-level control via request-response interactions. Amongst all, hybrid data acquisition aims to achieve the optimal performance for specific applications by combining multiple data acquisition modes.

Different types of data acquisition fit the needs of different application scenarios. Polling is favoured by energy-constrained networks due to its low complexity and predictability. In the past five years, polling schemes targeting energy efficiency, throughput, and fairness have been actively researched and applied to WSNs. Considering the fact that there is no polling scheme proposed for EM-WNSNs so far, the current research in

traditional WSNS is reviewed.

### 2.4.1 Polling in Battery-powered WSNS

In WSNS, one of the major sources of energy consumption is the idle listening time of nodes [132]. By reusing the idle listening time, the energy-efficient multi-hop polling [133] proposed for cluster-based WSNS prolongs network lifetime. In a collision-aware polling process, nodes causing collisions are scheduled to transmit data in separate time slots to avoid collision. As a tradeoff of low collision rate, nodes scheduled in future slots experience long idle time with unnecessary energy consumption. To maximally utilize idle time, an optimal polling schedule algorithm is adopted. In the optimal polling, the cluster head collects the network topology and let the nodes communicating with different receivers transmit in the same time slot to reduce the total polling duration. As expected, the proposed polling scheme shows high network lifetime, throughput and energy efficiency in multi-hop networks.

The priority-based polling scheme [134] for multi-hop cluster-based WSNS also aims to reduce the idle listening time for a few selected nodes in the network. In multi-hop networks, traffic generated by remote nodes turn the nodes nearby sinks into hot spots due to unbalanced energy consumption of forwarding. To prolong network lifetime, nodes located closer to the cluster head are prioritized for polling to achieve short idle time (these nodes are called key stations). In contrast, nodes far away from the cluster head are polled when key stations have no packets to transmit. Ultimately, the priority-based polling reduces the idle time of key stations therefore prolonging network lifetime.

### 2.4.2 Polling in Energy-harvesting WSNS

The above polling schemes are designed for WSNS powered by batteries and are inappropriate for Wireless Sensor Networks Powered by Ambi-

ent Energy Harvesting (WSN-HEAP) because of the dynamic often unpredictable energy states of nodes [135]. In WSN-HEAP, nodes respond to polling beacons with message packets only when they have sufficient energy. Due to the fluctuation of energy harvesting pattern, the first priority of data polling becomes network throughput. For this purpose, probabilistic polling [136,137] has been proposed.

In probabilistic polling, the sink broadcasts a polling beacon with a contention probability to acquire data from sensor nodes. The sensor nodes that receive the polling beacon generate their own random numbers that lies between 0 and 1. A sensor node responds to polling with message packets only when its own random number is smaller than the contention probability. The adjustment of contention probability follows the additive-increase multiplicative-decrease (AIMD) protocol that favours the “decrease” operation to avoid message collision. Probabilistic polling is shown to suit the characteristics of energy harvesting sensor network and yields high network throughput, fairness and low polling delay.

In the scenario where nodes harvest energy at different rates, nodes with faster charging speed have more opportunities to transmit thus creating unfairness. To solve this problem, a fair polling scheme based on probabilistic polling is proposed in [138]. For each polling, nodes respond to polling based on their locally modified contention probability which considers individual energy-harvesting rates. Specifically, nodes with low harvesting rate are prioritized to transmit message packets for the sake of fairness. From the evaluation results, the fair polling scheme shows high fairness without compromising much throughput.

### **2.4.3 Summary of Polling in WSNs**

According to the literatures reviewed and summarized in Table 2.4, polling schemes in WSNs do not consider the dynamic channel conditions and IoT bandwidth thus can hardly be applied for EM-WNSNs in the context

Table 2.4: Summary of Polling in WSNS

Protocol	Channel Condition	Energy Harvesting	IoT Bandwidth
Energy-efficient multi-hop polling [133]	Static	x	x
Priority-based polling [134]	Static	x	x
Probabilistic polling [136,137]	Static	√	x
Fair polling [138]	Static	√	x

of IoT.

## 2.5 Summary

Based on the literature review, it is concluded that there is a research gap in connecting EM-WNSNs to IoT. Few contributions in EM-WNSNs as well as WSNs handles data delivery in a way considering both the THz frequency selective feature and the limited IoT bandwidth. Therefore, solutions for data acquisition and dissemination are proposed in the following chapters to realize the concept of IoNT.

# Chapter 3

## Preliminary Study: CForward

In this chapter the preliminary work of the Channel-aware Forwarding (CForward) is presented as an initial step. As reviewed previously, EM-WNSNs are envisaged to operate in the THz band. The frequency selective path loss and noise temperature in the THz band are distance-dependent. Specifically, they get more severe with increasing transmission distance. Taking the advantage of this fact, a forwarding scheme that prioritises short-distance transmissions minimizes or even avoids the effect of frequency selective feature. However, if nodes consistently forward to the neighbours with shortest distance, the end-to-end hop count for the packet delivery will be large and this adversely affects the end-to-end performance. CForward is thus designed to overcome both the THz frequency selective feature and the multi-hop influence on network performance. Next, the end-to-end channel capacity and delay of CForward are evaluated against classical forwarding schemes in traditional WSNs.

### 3.1 Algorithm Design

The design of CForward for EM-WNSNs is presented. First, the assumptions are stated:

1. The locations of nanosensors are known by the nano-sink.
2. The nano-sink is responsible for the centralized operations, for example, determining the potential relay nodes of the sender when the sender intends to transmit.
3. All the nanosensors have sufficient energy to relay data.

In the THz band, the forwarding scheme that prioritises short-distance transmissions minimizes the effect of frequency selective feature. However, consistently forwarding to the neighbours with shortest distance adversely affects the end-to-end performance due to the high hop count [139]. Therefore, CForward is designed to make sure the data will not be forwarded to the relay node located: (i) in the long-distance region that is adversely affected by the frequency selective feature and (ii) in the short-distance region that will result in an unnecessarily large hop count.

For the node attempting to transmit, all the relay nodes within its transmission range are assigned a distance-based selection index  $I_S$  in Eqn. (3.1) by the nano-sink with sufficient energy and computation capacity, after which the relay node with the highest  $I_S$  will be chosen as the next-hop relay. The selection index is given by:

$$I_S = I_{Pri} - I_{Pen}, \quad (3.1)$$

whereby

$$I_{Pri} = 10 \log_{10} \left( \frac{C_{THz}}{C_{Ref}} \right), \quad I_{Pen} = \frac{1}{\sqrt{d+w}} \quad (3.2)$$

where  $d$  is the distance (in meters),  $w$  is the tuning factor of penalty index subject to  $\{w : 0 \leq w \leq 1\}$ ,  $C_{THz}$  is the THz channel capacity given by (2.2), and  $C_{Ref}$  is the reference channel capacity (neglecting molecular absorption) given by:

$$C_{Ref}(d) = \sum_{i=1}^n B \log_2(1 + SNR_{Ref}(f_i, d)) \quad (3.3)$$

$$SNR_{Ref}(f, d) = \frac{S(f)}{A_{spread}(f, d)N_0} \quad (3.4)$$

where  $f_i$  is the center frequency of the  $i$ -th sub-channel,  $B$  is the bandwidth of each sub-channel, and  $N_0$  is the minimum value of  $N_{abs}$  over the frequency band representing the constant noise P.S.D for the bandwidth  $B$ .

The index  $I_S$  penalizes selecting distant nodes that would result in higher path loss while  $I_{Pen}$  is designed to avoid the large hop count for repeatedly choosing short-distance transmission. The nearer the relay node is located, the larger the value of  $I_{Pen}$  will be, which leads to the decrease of  $I_S$ . The index  $I_{Pen}$  is tuneable via changing the value of  $w$ . Due to the distance-dependent frequency selective losses of the THz band, the farther the relay node is located, the more the THz channel capacity  $C_{THz}$  decreases. Therefore, the value of  $\left(\frac{C_{THz}}{C_{Ref}}\right)$  decreases leading to the decrease of  $I_S$ .

## 3.2 Performance Evaluation

In this section, the end-to-end performance of the forwarding schemes is evaluated and analyzed. Results in the figures are evaluated at discrete points with connecting lines to improve presentation.

### 3.2.1 Performance Metrics

Two metrics: 1) the end-to-end capacity; and 2) the end-to-end delay, are designed to evaluate the performance of CForward.

Due to interference, collision, and MAC mechanisms that introduce potential latency, the channel capacity experiences a certain amount of loss at each hop towards the sink [139, 140]. A simple end-to-end capacity model  $C_{e2e}$  in Eqn. (3.5) is designed to measure the reduced capacity due to losses

incurred at each hop en-route to the sink. In this model, the impact of interference on the per-hop capacity loss is a function of the average relative distance  $d_{AVG}$  between adjacent nodes given by  $\frac{1}{N} \sum_{i=1}^N d_i$ , with  $d_i$  denoting the distance between the sender and receiver nodes of the  $i$ -th hop. Specifically, if  $d_{AVG}$  is short, which means a dense network, the overall interference from other nodes at each hop is high. Conversely, in a network with low node density (large  $d_{AVG}$  value), the overall interference at each hop is low. The end-to-end capacity is expressed as:

$$C_{e2e} = C_I (1 - F_{AVG})^N, \quad (3.5)$$

where  $C_I$  is the channel capacity contributed by the first hop,  $N$  is the hop count determined by the forwarding scheme and  $F_{AVG}$  is the average capacity loss factor per hop. The value of  $F_{AVG}$  is calculated as follows:

$$F_{AVG} = F \left( \frac{d_o - d_{AVG}}{d_o} \right), \quad (3.6)$$

where  $F$  is the capacity loss factor and  $d_o$  is a constant denoting the reference distance from source-to-sink.

The end-to-end delay  $T_{e2e}$  is modelled assuming the absence of collision and channel contention (considering TS-OOK) and is indicative of the delay incurred with different forwarding schemes, as follows:

$$T_{e2e} = NT_{data} + T_{prop}^o \sum_{i=1}^N \left( \frac{d_i}{d_0} \right) \quad (3.7)$$

where  $T_{data}$  is the packet transmission time,  $T_{prop}^o$  is the source-to-sink propagation delay,  $d_i$  is the  $i$ -th hop distance, and  $N$  is the total hop count. The first term in Eqn. (3.7) expresses the processing delay at each of the  $N$  hops while the second term expresses the propagation delay.

### 3.2.2 Benchmarks

The following forwarding schemes in traditional WSNs are adopted as benchmarks:

1. The nearest forwarding (NForward) that selects the nearest neighbour as the next-hop forwarder.
2. The longest forwarding (LForward) that forwards to the furthest neighbour.
3. The random forwarding (RForward) that randomly selects one neighbour as its forwarder.
4. The single-hop end-to-end transmission.

### 3.2.3 Simulation Setup

A 1-m-long string network shown in Fig. 3.1a and 3.1b is used for the performance evaluation. The number of relay nodes varies from 5 to 50 with a step of 5 to form networks with different node densities. Fig. 3.1a shows a scenario whereby relay nodes are uniformly placed and Fig. 3.1b shows a scenario for randomly placed relay nodes between the source and the sink. All the nodes are assumed to be within the transmission range of one another and have enough energy to perform the packet forwarding. For this network the farthest distance between two nodes is 1 m. The 100-fs-long Gaussian pulse with 0.1 aJ energy [16] is transmitted over the entire THz band (0.1 THz — 10 THz). A packet consisting of 200 bits is transmitted from the source to the sink. The reference capacity loss factor  $F$  in Eqn. (3.6) is set to 0.4 while the tuning factor  $w$  in Eqn. (3.2) is set to  $1.2 \times 10^{-2}$ .



Figure 3.1: Network topology with ten nano nodes

For each scenario with random characteristics given by the random

forwarding scheme and the random node placement, the simulation is repeated 200 times. The results are presented with 95% confidence intervals.

For the performance comparison, firstly, the end-to-end performance is evaluated over different node densities (given by number of relay nodes) in channels with fixed water vapour content. The second set of simulations evaluate the end-to-end performance as a function of increasing water vapour content.

### 3.2.4 Varying Node Densities

In this section, the performance of the CForward is compared against the four classic forwarding schemes as the node density increases.

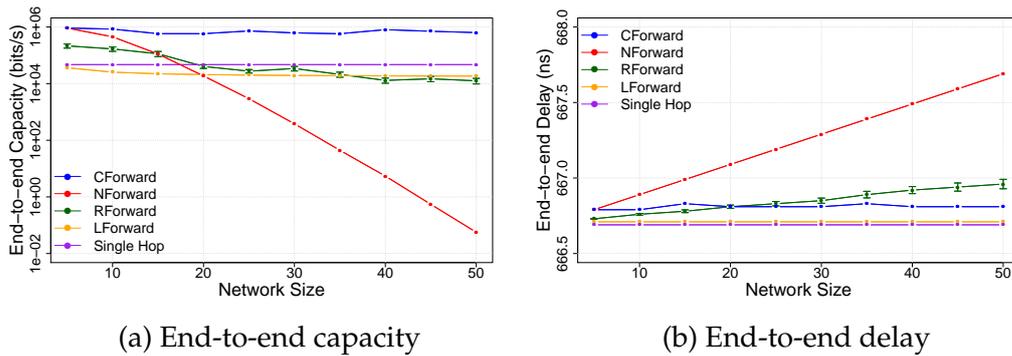
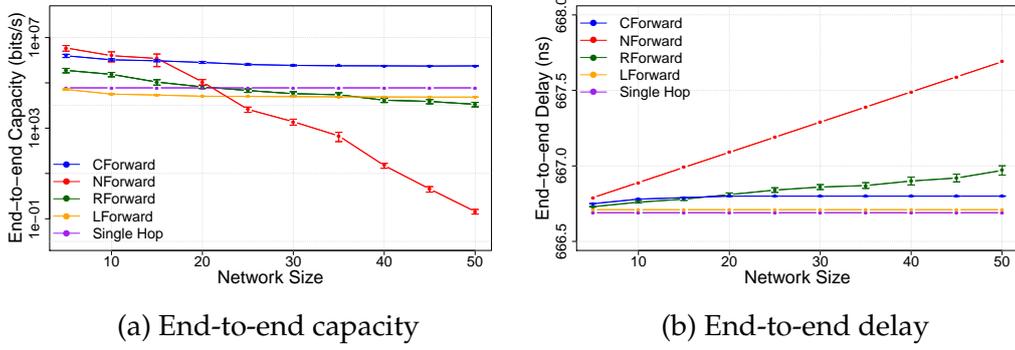


Figure 3.2: Performance for uniform topology with 10%  $H_2O$

For a channel containing 10%  $H_2O$ , the end-to-end capacity and delay of forwarding schemes are presented over an increasing number of relay nodes in Fig. 3.2a and 3.2b / Fig. 3.3a and 3.3b for the uniform/random node placement respectively. CForward shows the highest end-to-end capacity (741045.5 bps) and does not suffer huge end-to-end delay (666.801 ns), thus meeting the goal of the design.

The CForward makes forwarding decisions based on  $I_S$  therefore the forwarders selected are mostly close to the ideal relay located at a distance

Figure 3.3: Performance for random topology with 10% H<sub>2</sub>O

that generates an  $I_S$  of 1. This decision balances hop count and capacity, which makes the per-hop forwarding distance relatively deterministic. The denser the network is, the more likely a relay node appears on the ideal distance, leading to a more deterministic per-hop forwarding distance. As a result, the end-to-end performance of CForward, which shows an average confidence interval of 0.025ns, tends to be stable with increasing node density.

The simplest scheme, which is the single-hop transmission, shows the best delay performance but the channel capacity is restricted by the frequency selective effect. Multi-hop forwarding schemes show better capacity performance than the single-hop transmission in low-density networks but tradeoff delay for this improvement. Comparatively, CForward shows desirable capacity performance while maintaining low delay even in high-density networks. Compared with the single-hop transmission, CForward increases the end-to-end capacity by 1379.27%/1221.21% and introduces 0.018%/0.016% extra end-to-end delay for the uniform/random node placement.

The longest forwarding scheme has similar performance as single-hop transmission when the node density is low. Longest forwarding scheme only experiences two hops to reach the sink hence the end-to-end delay is low. However, the long transmission distance in the first hop penalizes

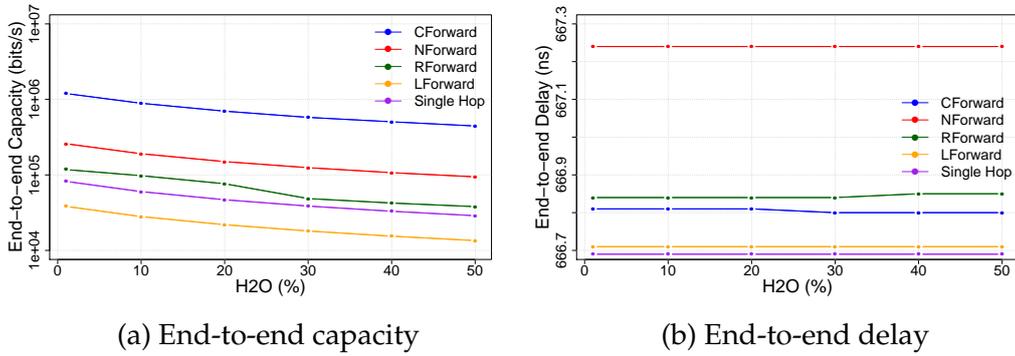
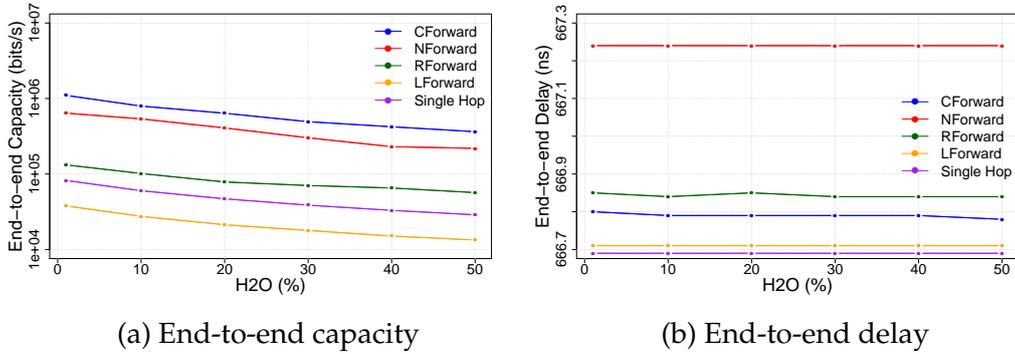
the capacity. As the node density increases, the first-hop transmission distance tends to be longer, meanwhile, the per-hop capacity loss gets larger, therefore the end-to-end capacity of the longest scheme gradually gets inferior to that of the single-hop transmission.

The nearest forwarding scheme is highly dependent of the node density of the network. It exhibits high end-to-end capacity only when the node density is low. This is contributed by two factors: (i) the short distance transmission that indicates high capacity and (ii) relatively small hop count. When the node density increases, the benefit of shorter transmission distance no longer offsets the detriment attributed to the larger hop count and the larger per-hop capacity loss given by Eqn. (3.6). Therefore, the end-to-end performance of capacity and delay both tend to be worse with increasing node density.

The random scheme selects different combinations of forwarding nodes. Results for the random scheme are presented as an average of 200 iterations. It is observed that the random scheme records higher hop count compared to CForward. This is because the denser the network is, the higher the expectation of residual hops would be, leading to higher end-to-end delay. Moreover, the end-to-end capacity and delay curves are sandwiched approximately between the nearest scheme and the longest scheme. This is because the random scheme on average tends to select forwarder in the middle between the nearest and the farthest. Thus, it has a smaller hop count than the nearest scheme and higher per-hop capacity than the longest scheme.

### 3.2.5 Varying H<sub>2</sub>O Percentage

The results of averaged end-to-end performance over an increasing percentage of H<sub>2</sub>O are shown in Fig. 3.4a and 3.4b / Fig. 3.5a and 3.5b for the uniform/random node placement respectively. Overall, the end-to-end capacity of all schemes decrease when the channel has higher H<sub>2</sub>O content,

Figure 3.4: Performance for uniform topology with varying H<sub>2</sub>OFigure 3.5: Performance for random topology with varying H<sub>2</sub>O

which is shown by an average decrease of 64.31% when water increases from 1% to 50%. For most schemes, the end-to-end delay is proportional to the hop count determined by specific forwarding schemes and is not directly influenced by the H<sub>2</sub>O content. However, this must be interpreted with the assumption that no collisions or retransmissions occur.

In terms of the end-to-end capacity, all the multi-hop forwarding schemes except the longest scheme outperform the single-hop transmission which is strongly affected by the frequency selective feature. The CForward shows the best end-to-end capacity performance with slightly decreased end-to-end delay over the increasing amount of H<sub>2</sub>O. It is inferred that the end-to-end delay serves as a tradeoff for the end-to-end capacity. Different for-

warding schemes take advantage of this tradeoff in different levels based on their forwarding mechanisms.

### 3.2.6 Discussion on the Optimal Case

To further investigate the performance of CForward, its end-to-end capacity is evaluated against the optimal solution that is implemented via exhaustive search. A comparison for both the linear topology and random topology is presented in Fig. 3.6. The results represent the mean and 95% confidence interval of all humidity levels as adopted in the previous simulations. It can be concluded that CForward with default parameters achieve acceptable end-to-end capacity and a more feasible searching process in comparison with the optimal solution. The future fine-grained parameter tuning for CForward is envisaged to minimize the gap between CForward and the optimal solution.

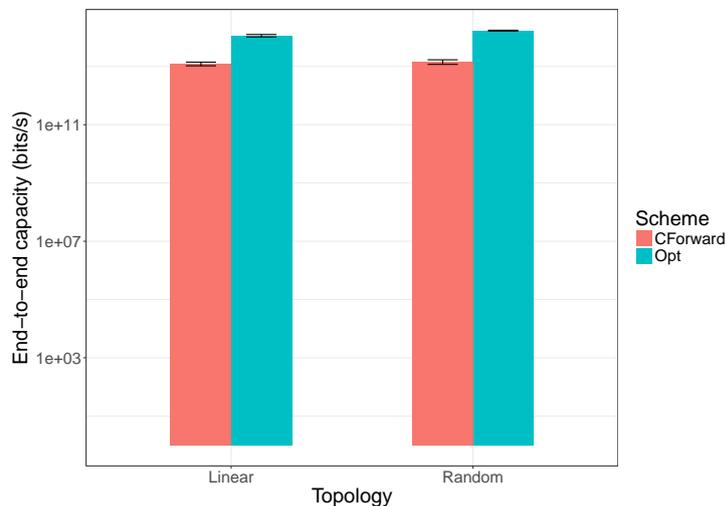


Figure 3.6: Comparison between CForward and the optimal

### **3.3 Summary**

The CForward presents high end-to-end channel capacity, which meets the goal of efficiently utilizing the THz band. Nevertheless, as an early-stage work, it lacks the consideration of the resource demand, dynamic channel conditions, as well as the limited IoT bandwidth, which limits the application scope.

# Chapter 4

## Bandwidth-aware Polling for IoNT

In the architecture of IoNT shown in Fig. 4.1, densely deployed nanosensors form the data tier that is responsible for generating sensed data which are aggregated by the backhaul tier composed of less dense nano-sinks. Nano-sinks then forward the aggregated data to an IoT gateway which will upstream data via the backhaul link connected to the Internet.

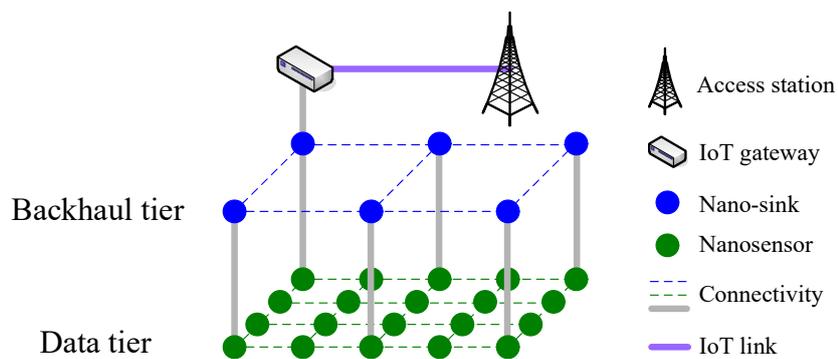


Figure 4.1: EM-WNSNs bridged to the IoT gateway

Data acquisition for IoNT faces two challenges: the dynamic IoT bandwidth and THz channel conditions that jointly impact resource utilization

efficiency. Specifically, the mismatch between the demands of the EM-WNSNs and the available bandwidth of the IoT link reduces the bandwidth efficiency of the IoT link or the energy efficiency of EM-WNSNs.

To achieve both high bandwidth efficiency of IoT infrastructure and high energy efficiency in EM-WNSNs, the On-demand Probabilistic (OP) polling (OP polling) is designed to acquire data from EM-WNSNs. In OP polling, the IoT gateway adjusts the packet aggregation based on the up-to-date information of IoT bandwidth and network connectivity to match the EM-WNSNs demand with the IoT bandwidth. Therefore, IoT bandwidth is efficiently utilized thus reducing energy consumption of nano-devices.

In this chapter, the optimal packet size for delay-constraint polling is firstly derived. Next, the design and evaluation of OP polling are presented.

## 4.1 Optimal Packet Size for Polling

In EM-WNSNs bridged to the IoT, the IoT bandwidth, i.e. bandwidth allocated to the IoT gateway by the WAN, defines the upper bound of effective traffic volume out from EM-WNSNs. Ideally, the IoT bandwidth should match the demand of EM-WNSNs. The mismatch between the bandwidth demand of EM-WNSNs and the IoT bandwidth leads to low bandwidth efficiency of the IoT link or low energy efficiency of EM-WNSNs.

On the one hand, unnecessary bandwidth provisioning caused by the EM-WNSNs demand mismatch reduces the utilization efficiency of the IoT bandwidth. On the other hand, insufficient IoT bandwidth provisioning results in extraneous packets that are discarded at the IoT gateway.

In time-critical sensing applications [141] (e.g. target tracking) it is better to discard than buffer packets, that soon become outdated, and reallocate resources to new sensor data. Hence, matching the demand to the IoT bandwidth saves energy in nano-devices by not processing data packets

that cannot be accommodated by the IoT link.

To further understand the low efficiency, the bandwidth demand of EM-WNSNs is denoted as  $T_{EM-WNSNs}$  and the IoT bandwidth as  $T_{BK}$  and both are defined as follows:

$$T_{EM-WNSNs} = \frac{p \sum_{i=1}^N n_i}{D}, \quad (4.1)$$

$$T_{BK} = \alpha T_{BK}^{max}, \quad (4.2)$$

where  $p$  is the size of one packet generated by a nanosensor,  $N \in [1, N_{max}]$  is the number of nano-sinks whose aggregated packets are received by the IoT gateway,  $N_{max}$  is the maximum number of nano-sinks connected to the IoT gateway,  $n_i \in [0, n_{max}]$  is the packet aggregation size denoting the number of packets aggregated at  $i$ -th nano-sink,  $n_{max}$  is the maximum packet aggregation size,  $D$  is delay constraint on the IoT link required by monitoring applications,  $\alpha \in [0, 1]$  is the ratio of bandwidth allocated to IONT and  $T_{BK}^{max}$  is the total bandwidth.

The OP polling is designed for delay-sensitive sensing applications that require data packets delivered within strict delay constraints [142]. From the above, the bandwidth demand of EM-WNSNs  $T_{EM-WNSNs}$  is the required bandwidth to transport data packets of EM-WNSNs within the application-specified delay constraint. The IoT bandwidth  $T_{BK}$  is defined as the portion  $\alpha$  of the total bandwidth  $T_{BK}^{max}$  offered by the IoT link as a result of the contention between H2H and M2M communications.

Consider the following two cases of mismatch:

$$T_{EM-WNSNs} < T_{BK}, \text{ where } n \neq n_{max}, N \neq N_{max}, \quad (4.3)$$

$$T_{EM-WNSNs} > T_{BK}. \quad (4.4)$$

When the IoT bandwidth allocated to the EM-WNSNs traffic exceeds the actual demand as expressed by Eqn. (4.3), the IoT bandwidth would be

under-utilized by the amount given by  $T_{BK} - T_{EM-WNSNs}$ . However, if the EM-WNSNs already generates the maximum data volume (i.e. when  $n = n_{max}$  and  $N = N_{max}$ ) this case is not regarded as poor efficiency.

In contrast, if the data volume of EM-WNSNs demands a IoT bandwidth that exceeds the IoT bandwidth as shown in Eqn. (4.4), the packets from the extra demand  $T_{EM-WNSNs} - T_{BK}$  would be discarded because they violate the delay constraint. In consequence, the energy efficiency of EM-WNSNs is reduced due to the unnecessary energy consumed in handling redundant packets.

The optimal resource efficiency is achieved when the bandwidth demand matches the IoT bandwidth, as shown in Eqn. (4.5)

$$T_{EM-WNSNs} = T_{BK}, \quad (4.5)$$

substituting Eqn. (4.1) into Eqn. (4.5), the optimal number of packets generated by EM-WNSNs (denoted by  $P_{opt}$ ), which indicates the maximum data volume that can be carried by the IoT link without compromising delay, is obtained and shown in Eqn. (4.6).

$$P_{opt} = \left\lfloor \frac{T_{BK} D}{p} \right\rfloor. \quad (4.6)$$

For each polling, if the number of packets polled from EM-WNSNs denoted by  $N_{EM-WNSNs}$  equals  $P_{opt}$ , there would be minimal to no residual IoT bandwidth or redundant data, leading to the optimal bandwidth efficiency of the IoT link and energy efficiency of EM-WNSNs.

To achieve this goal, OP polling is designed and evaluated in the following sections.

## 4.2 On-demand Probabilistic Polling

This section presents the assumptions of the application scenario and design of OP polling.

### 4.2.1 Assumptions

1. Both the channel condition and IoT bandwidth remain constant during one polling process which is composed of four handshakes. This assumption is reasonable considering the slow change rate of weather and macro-scale access and backhaul networks in comparison with the low latency of nanonetworks with small coverage [26, 143]
2. The IoT gateway knows  $T_{max}$  that is the longest delay from a sink to the gateway. This is obtained during network planning by calculating the maximum end-to-end delay when each sink is connected to its closest neighbour towards the gateway.
3. The IoT gateway has  $pn_{max}N_{total}$  bits storage, whereby  $N_{total}$  is the total number of sinks deployed, for the data polled.
4. During the polling process, nano-sinks are awake for message dissemination.

### 4.2.2 Algorithm Design

Matching the EM-WNSNs demand and IoT bandwidth faces two challenges: first, from the perspective of the IoT link, the bandwidth allocated to EM-WNSNs is dynamic and limited due to the contention between Machine-to-Machine (M2M) communications and Human-to-Human (H2H) communications [144]. Apart from that, from the perspective of EM-WNSNs, the bandwidth demand also shows a dynamic nature due to the humidity-sensitive THz channel that influences the network connectivity of the backhaul tier in EM-WNSNs. Consequently, the dynamic nature of both the EM-WNSNs demand and the IoT bandwidth reduces the utilization efficiency of network resources.

The OP polling achieves both high bandwidth efficiency of IoT link and high energy efficiency of EM-WNSNs by concurrently taking into consideration the dynamics of both IoT bandwidth and THz channel condi-

tions. In OP polling, the IoT gateway polls nano-sinks which then poll nanosensors for data extraction. For each polling event, the IoT gateway dynamically adjusts the packet aggregation size of each nano-sink, which is denoted as  $n \in [0, n_{max}]$  whereby  $n_{max}$  is the maximum aggregation size considering the limited energy capacity of nano-sinks, based on the queried IoT bandwidth before pushing data into the IoT link in order to match the EM-WNSNs demand with IoT bandwidth. OP polling aims to achieve high bandwidth efficiency by utilizing the residual bandwidth.

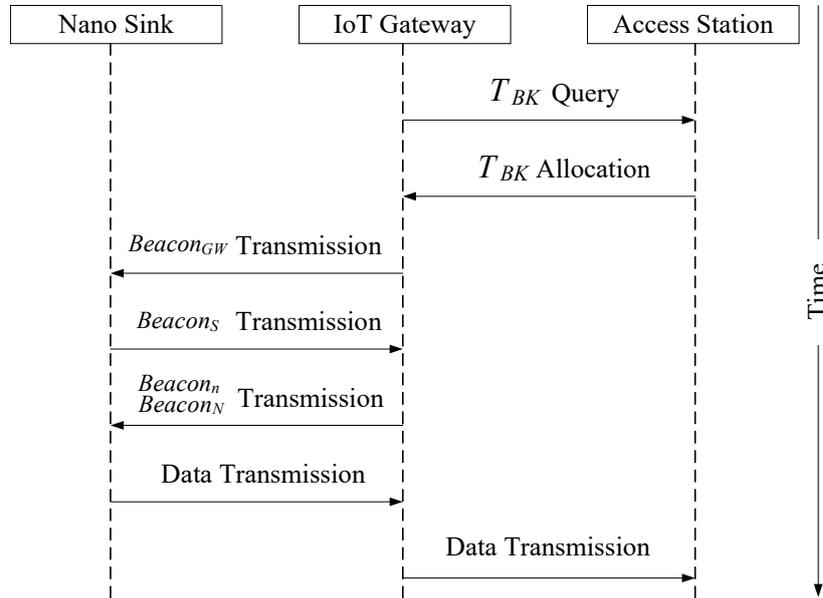


Figure 4.2: Sequence of message exchange in OP polling

The operations of OP polling for the IoT gateway and nano-sinks are presented in Algorithm 1 and Algorithm 2, respectively. The notations for algorithm operations are listed in Table 4.1. The responding operations of nano-sinks are presented via Fig. 4.2 which depicts the sequence of message exchange of the overall polling process. OP polling requires control processes on packet aggregation before polling. At the start of each polling cycle, the IoT gateway queries the allocated  $T_{BK}$  from access stations such as eNodeB in LTE (cf: Algorithm 1 and Fig. 4.2) and broadcasts a gateway

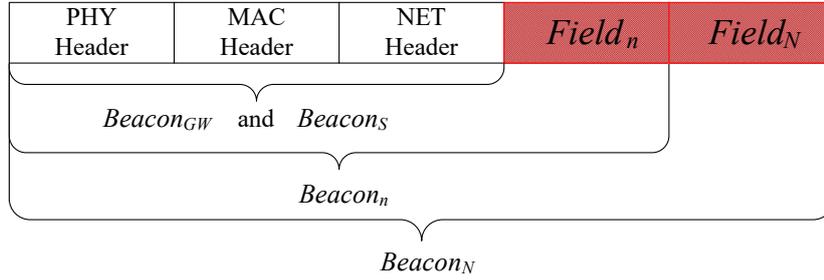


Figure 4.3: Packet structure of beacons

beacon  $Beacon_{GW}$  (Fig. 4.3) to obtain the maximum number of nano-sinks connected to the IoT gateway  $N_{\max}$ . Each sink that receives  $Beacon_{GW}$  forwards the beacon to neighbours and replies the IoT gateway with a sink beacon  $Beacon_S$  (Fig. 4.3) following line 2 of Algorithm 2.

After obtaining  $N_{\max}$ , if there is any nano-sink connected to the IoT gateway, the packet aggregation control shown in line 4 of Algorithm 1 will be executed for achieving high resource utilization efficiency. The packet aggregation control aims to adjust  $T_{EM-WNSNs}$  to match  $T_{BK}$  by dynamically controlling the packet aggregation size  $n$  based on  $N_{\max}$  and  $T_{BK}$ . Specifically, to ensure fairness during data acquisition process, the IoT gateway evenly allocates the available IoT bandwidth to all nano-sinks reachable by adjusting  $n$ . After that, following lines 5 - 8 of Algorithm 1,

Table 4.1: Notations used to describe OP polling

Notation	Description
$T_{BK}$	The allocated IoT bandwidth
$P_{qt}$	The optimal packet size for polling
$N_{\max}$	The maximum number of sinks connected to the gateway
$Beacon_{GW}$	The network discovery beacon
$Beacon_n$	The deterministic polling beacon
$Beacon_N$	The probabilistic polling beacon
$Field_n$	The deterministic packet size
$Field_N$	The probabilistic packet size

**Algorithm 1** On-demand Probabilistic Polling for IoT gateway

---

```

1: Query  $T_{BK}$  from the access station and calculate  $P_{qt}$ 
2: if  $P_{qt} > 0$  then
3:   Broadcast  $Beacon_{GW}$  to obtain  $N_{max}$ 
4:   if  $N_{max} > 0$  then
5:     Calculate  $n = \left\lfloor \frac{P_{qt}}{N_{max}} \right\rfloor$ 
6:     if  $P_{qt} - nN_{max} == 0$  or  $n == n_{max}$  then
7:       Broadcast  $Beacon_n$  with  $Field_n = n$ 
8:       Collect and send all data packets to the IoT link
9:     else
10:      Calculate  $N = P_{qt} - nN_{max}$ 
11:      if  $\left\lfloor 100 \frac{N}{N_{max}} \right\rfloor > 0$  then
12:        Broadcast  $Beacon_N$  with  $Field_n = n, Field_N = \left\lfloor 100 \frac{N}{N_{max}} \right\rfloor$ 
13:        Collect and send  $P_{qt}$  data packets to the IoT link
14:        Drop redundant packets
15:      end if
16:    end if
17:  end if
18: end if

```

---

if the up-to-date IoT bandwidth is maximally utilized or the calculated  $n$  reaches its maximum, the value of  $n$  will be inserted into the deterministic polling beacon  $Beacon_n$  shown in Fig. 4.3 that is then broadcast to nano-sinks. Each nano-sink  $i$  that receives  $Beacon_n$  aggregates and transmits  $n$  packets based on the value carried by the deterministic packet size  $Field_n$ , as shown in line 4 of Algorithm 2. Here, it is assumed that the data tier always has sufficient number of unique packets to be collected by nano-sinks considering the high density of nanosensors deployed [123]. After data collection, the IoT gateway directly forwards all data packets to the IoT link since in this case there will be no redundant demand exceeding IoT bandwidth.

**Algorithm 2** OP Feedback for Nano-sink  $i$ 


---

```

1: if receive a new  $Beacon_{GW}$  then
2:   Reply with  $Beacon_S$ 
3: else if receive a new  $Beacon_n$  then
4:   Aggregate and transmit  $n$  message packets
5: else if receives a new  $Beacon_N$  then
6:   Generate a random integer  $r^i \in [0, 100]$ 
7:   if  $r^i < Field_N$  then
8:     Aggregate and transmit  $n + 1$  message packets
9:   else if  $n \neq 0$  then
10:    Aggregate and transmit  $n$  message packets
11:   end if
12: end if

```

---

For OP polling, if the calculated  $n$  leads to residual IoT bandwidth that is sufficient for at least one extra packet, the IoT gateway will select  $N$  out of  $N_{\max}$  nano-sinks as in line 12 of Algorithm 1 to evenly share the residual bandwidth. Specifically, each of those  $N$  nano-sinks will aggregate  $n + 1$  instead of  $n$  packets. In order not to compromise fairness, the concept of probabilistic polling is adopted for selecting sinks that utilize residual bandwidth. Specifically, a contention probability of  $\frac{N}{N_{\max}}$  and the value of  $n$  are inserted to  $Field_N$  that denotes the probabilistic packet size and  $Field_n$  of the probabilistic polling beacon  $Beacon_N$ , respectively. Next,  $Beacon_N$  is transmitted to nano-sinks. Considering the limited compute and storage capacity of nano-sinks, the value of  $\frac{N}{N_{\max}}$  is converted to an integer  $\left\lfloor 100 \frac{N}{N_{\max}} \right\rfloor$  that lies in  $[0, 100]$ .

Following lines 5 - 12 of Algorithm 2, when a sink  $i$  receives  $Beacon_N$ , it generates a random integer  $r^i \in [0, 100]$  and compares  $r^i$  with  $\left\lfloor 100 \frac{N}{N_{\max}} \right\rfloor$ . If  $r^i < \left\lfloor 100 \frac{N}{N_{\max}} \right\rfloor$ , the sink will aggregate  $n + 1$  message packets. Otherwise  $n$  message packets will be aggregated. Lastly, in case that the probabilistic process polls more packets than desired, the IoT gateway only pushes  $P_{qt}$

aggregated packets to the IoT link with redundant packets dropped, as demonstrated in lines 13 and 14 of Algorithm 1.

It is to be noted that after transmitting each beacon ( $Beacon_{GW}$ ,  $Beacon_n$ , or  $Beacon_N$ ), the gateway has to wait for a short period  $T_{GW}$  to collect the response from sinks. This parameter is jointly determined by the delay constraint, network topology and corresponding routing solutions. In this thesis, a conservative waiting duration  $T_{GW} = T_{max}$  is adopted.

For implementation, OP polling is very suitable for simple low capacity devices that fit the characteristics of nano-devices. Beacon packets  $Beacon_{GW}$ ,  $Beacon_S$ ,  $Beacon_n$  and  $Beacon_N$  are lightweight with low overheads and the packet aggregation control is implemented using only two extra fields, which are  $Field_n$  of  $\lceil \log_2 n_{max} \rceil$  bits carrying  $n$  and  $Field_N$  of seven bits storing the integer between 0 and 100.

## 4.3 Performance Evaluation

In this section, the performance metrics and benchmark polling schemes are firstly discussed. Then, the simulation parameters are listed. Lastly, the results of performance evaluation are analysed.

### 4.3.1 Performance Metrics

The OP polling is evaluated using the following performance metrics:

1. Packet aggregation size  $n$ . It shows the effect of packet aggregation control algorithm in OE polling. To a large extent, the value of  $n$  is decisive for the performance metrics presented below.
2. IoT Bandwidth efficiency,  $BE$ :

$$BE = \frac{P_{EM-WNSNs}}{P_{qt}}, \quad (4.7)$$

where  $P_{EM-WNSNs}$  is the total number of packets pushed into the IoT link and  $P_{qt}$  is the optimal number of packets mentioned in Section

4.1. The  $BE$  of 100% represents the optimal utilization efficiency of IoT bandwidth.

3. Packet drop ratio,  $P_{drop}$ :

$$P_{drop} = \frac{P_{total} - P_{opt}}{P_{total}}, \quad (4.8)$$

where  $P_{total}$  is the total number of packets collected by the IoT gateway. The packet drop ratio indicates the volume of redundant packets dropped by the IoT gateway as packets are sent from the EM-WNSN to the Internet. This metric reflects the unnecessary resource consumption on nano-sinks.

4. Energy consumption of polling one packet,  $E$ :

$$E = \frac{E_{total}}{P_{EM-WNSNs}}, \quad (4.9)$$

where  $E_{total}$  is the total energy consumption of packet processing during one polling process. Specifically, the energy consumed by receiving each packet is 10% of the amount consumed by packet transmission [24].

5. Polling delay,  $L$ :

$$L = L_P + L_T, \quad (4.10)$$

where  $L_P$  and  $L_T$  are the propagation delay determined by distance and transmission delay related to packet length, respectively.

6. Polling coverage,  $C$ :

$$C = \frac{N}{N_{total}}, \quad (4.11)$$

where  $N$  is the number of nano-sinks that reply to polling with aggregated packets and  $N_{total}$  is the total number of nano-sinks in the network. Polling coverage evaluates the data integrity for sensing applications. However, under limited IoT bandwidth, high values of polling coverage do not always ensure good integrity of data due to packet dropping.

7. Collision probability,  $P_{COLL}$ :

$$P_{COLL} = \frac{N_{COLL}}{N_{RX}}, \quad (4.12)$$

where  $N_{COLL}$  is the total number of collision and  $N_{RX}$  is the total number of packet receptions during the end-to-end data delivery for one polling interval.

8. Fairness defined by the Jain's fairness index,  $F$ :

$$F = \frac{(\sum_{i=1}^N T_i)^2}{N \sum_{i=1}^N T_i^2}, \quad (4.13)$$

where  $T_i$  is the throughput of message packets from the  $i$ -th nano-sink during one simulation. Fairness provides a way to check if all nodes are equally selected by the probabilistic process of OP polling without preference.

### 4.3.2 Benchmarks

Considering that there has been little (if any) research on the scenario with dynamic IoT bandwidth and channel conditions, OE polling and greedy polling are adopted to benchmark OP polling.

1. OE polling. OE polling is similar to OP polling however without the probabilistic process. The operation sequence, packet structure, and basic operations of OE polling are presented in Fig. 4.4, Fig. 4.5, Algorithm 3 and Algorithm 4. To understand the trade-off between the bandwidth efficiency of the IoT infrastructure and the energy efficiency of EM-WNSNs, the packet aggregation size  $n$  in OE polling is calculated using either the floor function  $\left\lfloor \frac{P_{qt}}{N_{\max}} \right\rfloor$  or the ceiling function  $\left\lceil \frac{P_{qt}}{N_{\max}} \right\rceil$ . The resulted schemes are then called OE-floor polling and OE-ceil polling, respectively.

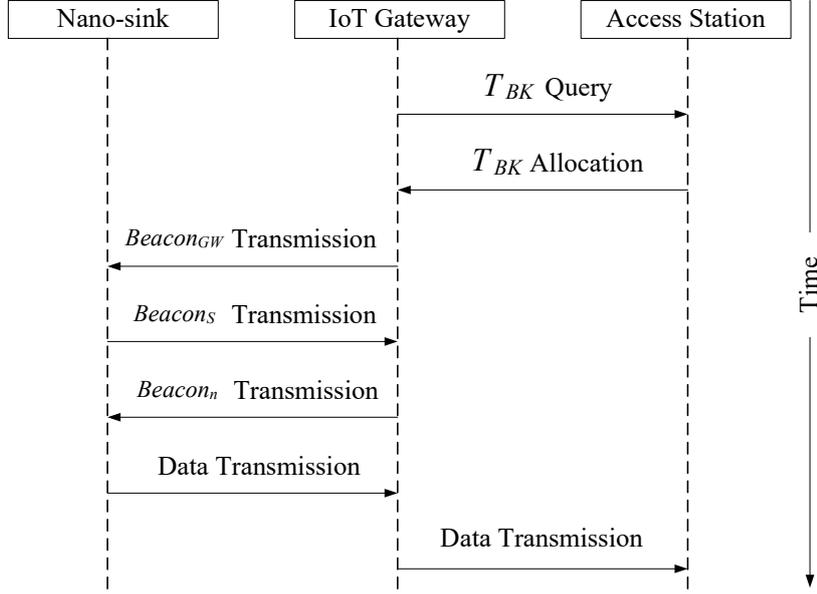


Figure 4.4: Sequence of message exchange in OE polling  
 $Beacon_{GW}$  and  $Beacon_s$

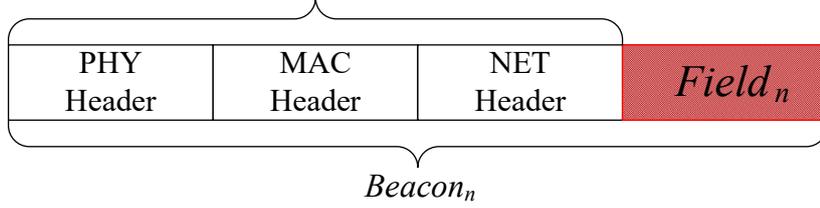


Figure 4.5: Packet structure of beacons in OE polling

2. Greedy polling. Greedy polling relies on the central process of IoT gateway to deliver data within delay constraints. First, the IoT gateway polls all nano-sinks by flooding a gateway polling beacon  $Beacon_{GW}$ . In response, each nano-sink polled aggregates and transmits  $n_{\max}$  packets with best effort. At last, similar to OP polling, the IoT gateway sends  $P_{qt}$  packets to the IoT link with the rest of packets dropped unless the IoT bandwidth is sufficient for all packets polled. In comparison with OP polling and OE polling, the greedy polling approach is optimized for the IoT bandwidth efficiency, at the expense of other performance metrics.

---

**Algorithm 3** On-demand Efficient Polling for IoT Gateway

---

- 1: Query  $T_{BK}$  and calculate  $P_{opt}$
  - 2: **if**  $P_{opt} > 0$  **then**
  - 3:   Broadcast  $Beacon_{GW}$  to obtain  $N_{max}$
  - 4:   **if**  $N_{max} > 0$  **then**
  - 5:     Calculate  $n = \lfloor \frac{P_{opt}}{N_{max}} \rfloor$  (OE-floor Polling)
  - 6:     or  $n = \lceil \frac{P_{opt}}{N_{max}} \rceil$  (OE-ceil Polling)
  - 7:     **if**  $n > 0$  **then**
  - 8:       Broadcast  $Beacon_n$  with  $Field_n = n$
  - 9:       Collect and transport all data packets to the IoT link
  - 10:     **end if**
  - 11:   **end if**
  - 12: **end if**
- 

---

**Algorithm 4** OE Feedback for Nano-sink  $i$ 

---

- 1: **if** receive a new  $Beacon_{GW}$  **then**
  - 2:   Reply with  $Beacon_S$  (respond to  $Beacon_{GW}$ )
  - 3: **else if** receive a new  $Beacon_n$  **then**
  - 4:   Aggregate and transmit  $n$  message packets
  - 5: **end if**
- 

### 4.3.3 Data Dissemination for the Polling System

To efficiently implement multi-hop polling and data dissemination under dynamic channel conditions, a Time-to-Live (TTL)-based Efficient Flooding (TEFlood) with low implementation complexity is used in all the polling schemes. The pseudo code of TEFlood is shown in Algorithm 5.

$TTL_{max}$  is a predefined network parameter for the longest possible distance (hop count) between the IoT gateway and farthest nano-sinks. When the IoT gateway broadcasts a beacon, it sets the (initial) value of  $TTL_P$  to  $TTL_{max}$ , where  $TTL_P$  is the TTL field for an arbitrary packet traversing the network. When a nano-sink  $i$  receives a new  $Beacon_{GW}$  from the IoT

---

**Algorithm 5** TTL-based Efficient Flooding for Nano-sink  $i$ 

---

```

1: if receive a new  $Beacon_{GW}$  or  $Beacon_n$  then
2:    $TTL_S^i = TTL_{max} - TTL_P$ 
3:   Rebroadcast packet with  $TTL_P = TTL_P - 1$ 
4: else if receive a new data packet or  $Beacon_S$  then
5:   if  $TTL_S^i < TTL_P$  then
6:     Rebroadcast packet with  $TTL_P = TTL_P - 1$ 
7:   end if
8: end if

```

---

gateway, it computes its own distance to the gateway (line 2: Algorithm 5) and assigns that value to  $TTL_S^i$ .

In this way, nano-sinks dynamically tune their own TTL settings  $TTL_S^i$  based on the TTL values of the received beacons to constrain the forwarder sets to those closer to the IoT gateway.

Nano-sinks respond to each polling by the IoT gateway with  $Beacon_S$  and data packets that have  $TTL_P$  field set to values close to the minimum hopcount to the IoT gateway. When a nano-sink receives a packet to be forwarded, it checks to ensure that it is closer to the IoT gateway (line 5) before forwarding it (line 6). This directed flooding approach is primarily aimed to keep the forwarding process as simple as possible while minimizing the network traffic and unnecessary data packet transmissions.

#### 4.3.4 Simulation Parameters

The performance of OP polling, OE polling, and greedy polling are evaluated using the Nano-Sim [145] tool on NS-3 with simulation parameters listed in Table 4.2. For both nano-sinks and the IoT gateway, each data bit is modulated using the Time Spread On-Off Keying (TS-OOK) [24] of one 100-fs-long pulse on 1 THz with 1 pJ energy. The receiving sensitivity of nano-devices is set to -130 dBm considering the high sensitivity of nano

materials [6]. Each unit message packet generated by a nanosensor contains 100 bits. The size of  $Beacon_{GW}$  and  $Beacon_S$  are set to 36 bits. For all packets, the ratio of bit "1" is assumed to be 50% [24]. Considering the energy capacity of nano-sinks, the maximum packet aggregation size is set to 10 packets that result in a 4-bit-long  $Field_n$ .

Table 4.2: Simulation parameters

Parameter	Value
Frequency	1 THz
Pulse energy	1 pJ / 10 aJ
Pulse duration	100 fs
Pulse interval	10000 fs
Receiving sensitivity	-130 dBm
$n_{\max}$ size	10 packets
$Field_n$ size	4 bits
$Field_N$ size	7 bits
$p$	100 bits
$Beacon_{GW}, Beacon_S$ size	36 bits
$Beacon_n$ size	40 bits
Polling interval	60 s / 1 h
$T_{BK}^{\max}$	20 Mbps
$D$	1 ms
$TTL_{\max}$	100
Simulation duration	600s / 24h
Simulation area	10m $\times$ 10m
Network size	10,20,30,40,50,60,70,80,90,100
IoT gateway position	Centre
Nano-sink position	Uniformly distributed
Nano-sink mobility	Static
$\alpha$	[0,1]
Molecular absorption coefficient $K$	[0.38,7.35] $m^{-1}$ / [0.11, 0.48] $m^{-1}$

The IoT link with total bandwidth  $T_{BK}^{\max}$  of 20Mbps is adopted to serve the polling process with delay tolerance  $D$  of 1ms. For node placement, from 10 to 100 static nano-sinks are randomly deployed with uniform distribution in a  $10m \times 10m$  area whereby an IoT gateway is located in the centre.

For each polling process, a random decimal  $\alpha \in [0, 1]$  is generated to implement the dynamic IoT bandwidth. The averaged value of  $\alpha$  in simulations is presented in Fig. 4.6a and Fig. 4.6b. The dynamic channel condition is implemented by a time-varying molecular absorption coefficient  $K$  in Eqn. (2.1). Specifically, two scenarios are considered for the time-varying channel. The first scenario simulates a Large-Scale-Varying (LSV) channel by adopting a half-cycle sinusoidal curve for every minute  $K$  where the minimum and maximum water vapour are 1% and 20% respectively, as shown in Fig. 4.6a. In the second scenario, hourly  $K$  with Small-Scale Variance (SSV), which represents a mix ratio of water from 0.29% to 1.23%, is generated by the CRU weather generator based on a first-order Markov chain model [146], as shown in Fig. 4.6b. These two channel conditions represent different stages of humidity change. For example, SSV channel represents the humidity variance in relatively dry environment where water vapour is contained in the air whereas LSV channel represents the humidity level when water vapour becomes rainfall. For LSV channel and SSV channel, each simulation lasts for 600s with 60s as the polling interval and 24 hours with 1 hour as the polling interval, respectively.

Each result presented is the averaged value with 95% confidence interval under normal error distribution obtained from 50 simulations with different seed values.

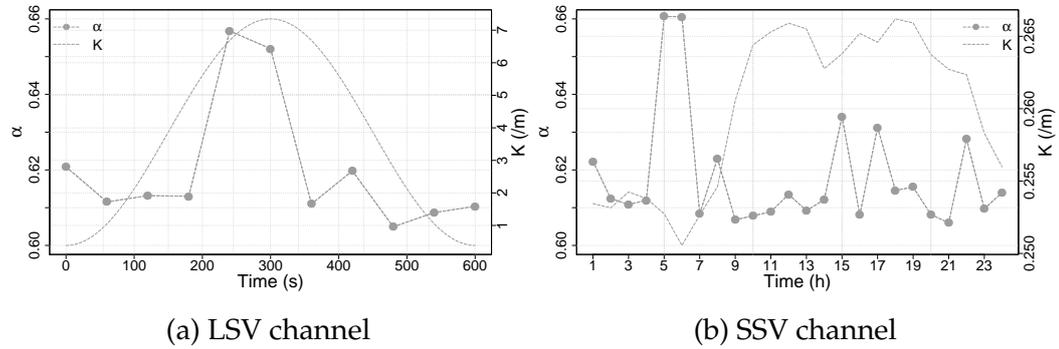


Figure 4.6: IoT bandwidth capacity ratio and channel state vs time

### 4.3.5 Results and Analysis

Simulation results for the performance metrics in section 4.3.1 are shown in Fig. 4.7a-4.22b over the simulation time slots and for different network sizes. The curve of  $K$  value is plotted within the results with time lines. Finally, the averaged results for LSV channel and SSV channel are summarized in Table 4.3 and Table 4.4, respectively.

#### Packet Aggregation Size

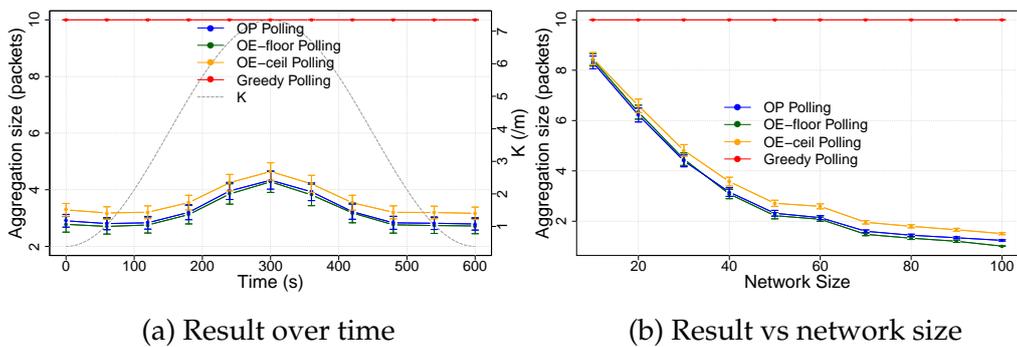


Figure 4.7: Packet aggregation size for LSV channel

The packet aggregation size  $n$  over time and different network sizes for LSV channel and SSV channel are presented in Fig. 4.7a - 4.8b. For results

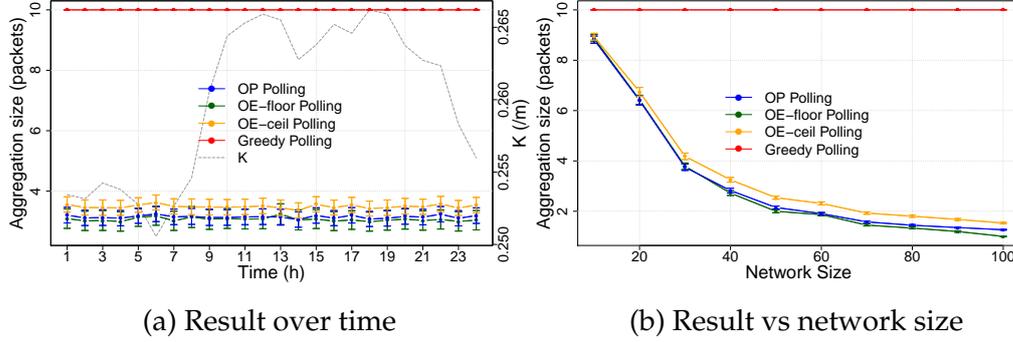


Figure 4.8: Packet aggregation size for SSV channel

over time, LSV channel shows significant variance whereas SSV channel does not due to the small percentile of molecular absorption loss over the total path loss.

The packet aggregation size  $n$  in OP polling and OE polling is dynamically adjusted according to the impact of IoT bandwidth and channel conditions represented by  $\alpha$  and  $K$ . Specifically, each nano-sink aggregates more packets for high IoT bandwidth efficiency when there is sufficient IoT bandwidth and sparse nano-sinks connected to the IoT gateway, which is a result of high  $\alpha$ , high  $K$  and small network size.

On the other hand, when the IoT bandwidth is limited, fewer packets are aggregated and there is strong contention among the nano-sinks. OP polling shows a larger aggregation size than OE-floor polling because of the probabilistic process that polls a subset of all nodes to aggregate  $n + 1$  packets to maximize the bandwidth efficiency. OE-ceil polling has a larger aggregation size than OP polling since each of all sinks mostly aggregates one packet more than OE-floor polling. In comparison, for greedy polling,  $n = n_{\max}$  for every polling event regardless of the change in  $\alpha$  and  $K$ .

### IoT Bandwidth Efficiency

By dynamically adjusting the packet aggregation size, OP polling is able

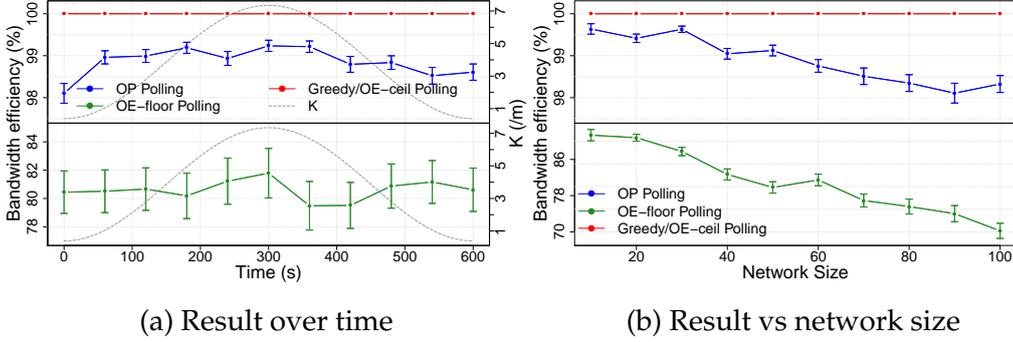


Figure 4.9: Bandwidth efficiency for LSV channel

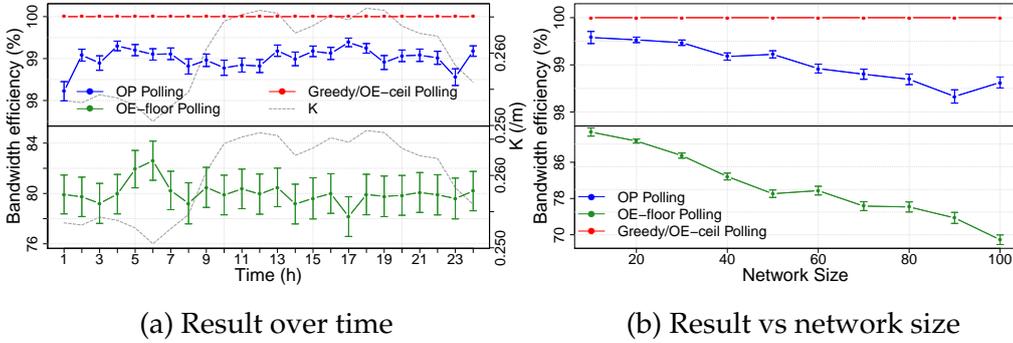


Figure 4.10: Bandwidth efficiency for SSV channel

to utilize almost all the available IoT bandwidth, on average 98.9416% (as shown in Fig. 4.9a - 4.10b) without incurring unnecessary overheads in energy consumption and packet loss, as shown below. OE-floor polling achieves lower bandwidth efficiency of 80.5270% on average, mainly due to the inability to utilize the residual bandwidth. The bandwidth efficiency of OP polling and OE-floor polling is positively correlated with  $\alpha$  and negatively correlated with network connectivity. On the one hand, high  $\alpha$  increases bandwidth efficiency because of the increased  $P_{opt}$  which reduces the percentile of unutilized bandwidth for each packet. On the other hand, although high connectivity resulting from small  $K$  or large network size reduces the bandwidth loss for individual users by reducing

$n$ , it increases the probability of collision that leads to unsuccessful packet delivery, which deteriorates bandwidth utilization. Greedy polling, as the name implies, as well as OE-ceil polling, fully utilizes the available IoT bandwidth, but at the expense of poor energy efficiency.

### Packet Drop Ratio

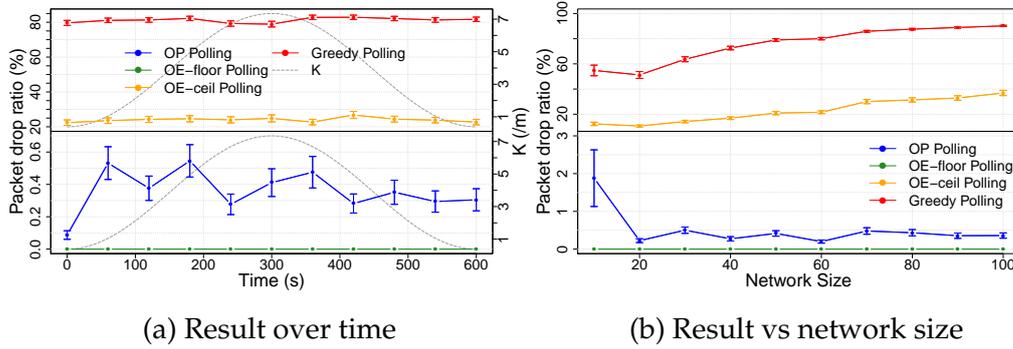


Figure 4.11: Packet drop ratio for LSV channel

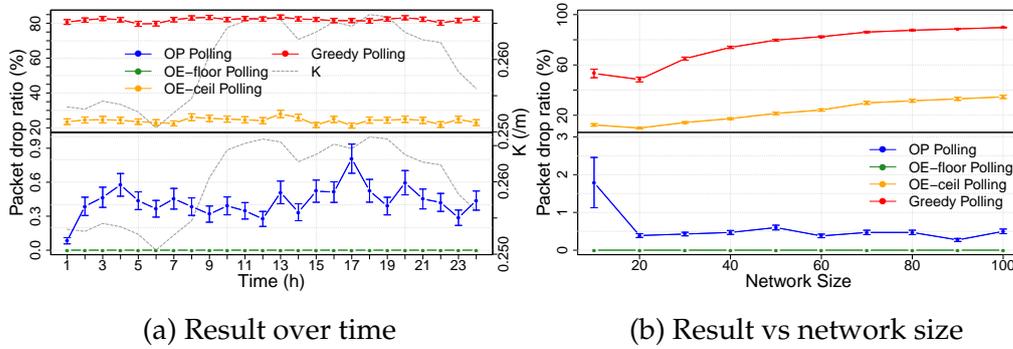


Figure 4.12: Packet drop ratio for SSV channel

Packet drop ratio is shown Fig. 4.11a - 4.12b. Although the throughput of OP polling shows probabilistic nature, only 0.4533% of the packets polled are dropped on average since the number of packets polled approaches  $P_{qt}$  which minimizes the number of redundant packets. In

comparison, OE-ceil polling exhibits a higher drop ratio of 23.6036% because that the packet size after ceiling may exceed the bandwidth limit. Greedy polling has the highest drop ratio of 79.1728% because all nano-sinks polled persistently aggregate the maximum number of packets regardless of IoT bandwidth limit. OE-floor polling, on the other hand, drops no packet because of its conservative polling strategy and the bandwidth is mostly under-utilized.

### Energy Consumption

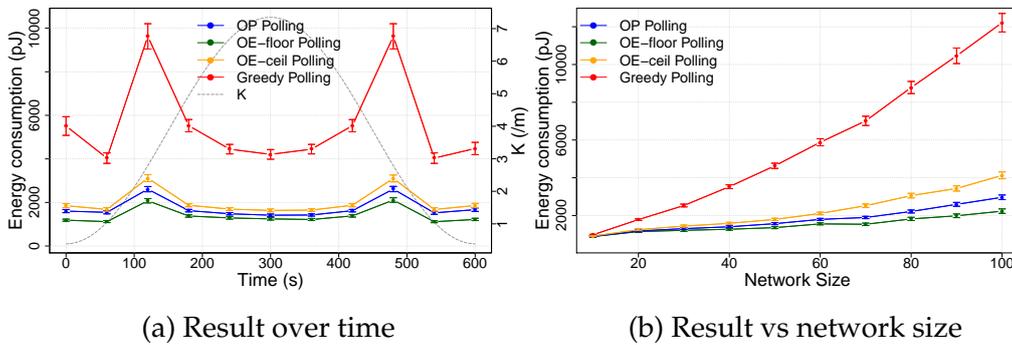


Figure 4.13: Energy consumption for LSV channel

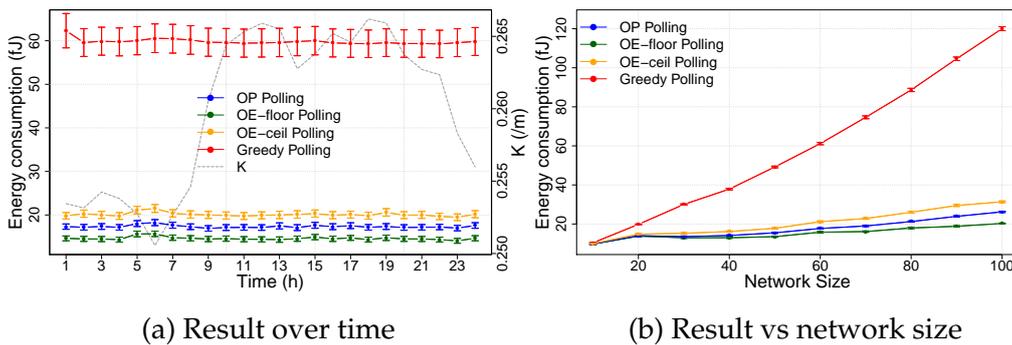


Figure 4.14: Energy consumption for SSV channel

Fig. 4.13a - 4.14b show that OP polling and OE polling successfully reduce the energy consumption for polling one packet compared to greedy polling. This is achieved by dynamically tuning packet aggregation size to match the EM-WNSNs' traffic demand with IoT bandwidth. For OP polling, because of the slightly higher packet drop ratio, higher energy consumption per packet as compared to OE-floor polling is observed. OE-ceil polling consumes more energy than OP polling because of the extra packets polled although it is much more efficient than greedy polling with no bandwidth-aware adjustment. For nano-devices with low energy capacity, the reduced energy consumption of OP and OE polling will improve network life time. For LSV channel, the energy consumption, especially for greedy polling, at 120s and 480s show two peaks as a result of the slightly decreased node density with increased end-to-end hop count that involve relaying across more sinks in data delivery.

### Polling Coverage

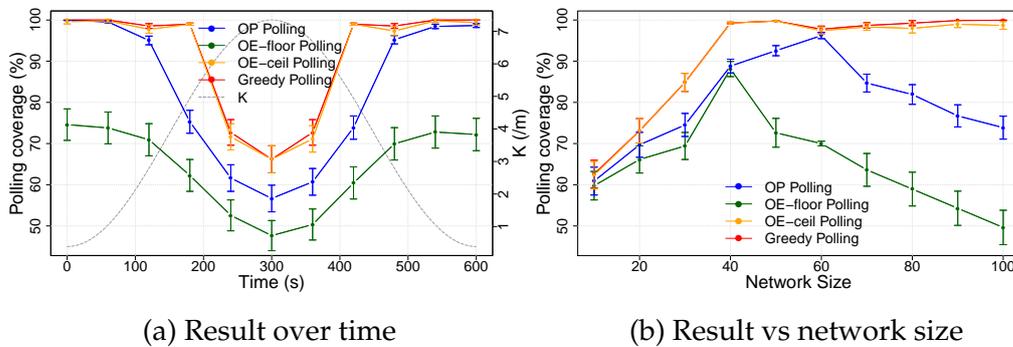


Figure 4.15: Polling coverage for LSV channel

As shown in Fig. 4.15a - 4.16b, OP polling shows a polling coverage that is sandwiched between OE-floor polling and OE-ceil polling. This corresponds to the tradeoff between high bandwidth efficiency and low packet drop ratio. OE-floor polling with the smallest coverage results

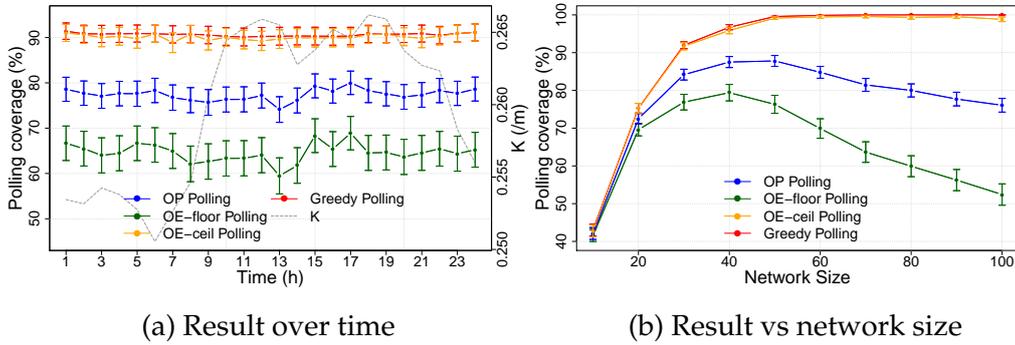


Figure 4.16: Polling coverage for SSV channel

in under-utilized IoT bandwidth, i.e. lowest bandwidth efficiency and also lowest packet drop ratio. Greedy polling achieves the best coverage among the three; however a large number of nano-sinks are unnecessarily polled giving rise to the high packet drop ratio. It is worth noting that greedy polling covers slightly more sinks than OE-ceil polling since the gateway is not aware of the bandwidth condition and it polls even when there is no bandwidth, which is a rare event. For LSV channel, polling coverage is high for small  $K$  and large network size whereby more nano-sinks can be polled. When network size increases beyond certain thresholds (60 for OP polling and 40 for OE polling), the polling coverage for OP polling and OE polling start to reduce as a result of the increased number of nodes that are not polled due to the bandwidth limit of the IoT link. However this behaviour is not observed for SSV channel due to the  $K$  values with low magnitude and variance.

### Polling Delay

As shown in Fig. 4.17a and Fig. 4.18b, OP polling and OE polling have higher delay than greedy polling due to the extra overheads caused by handshakes required for collecting network information. The delay of OP polling is slightly lower than that of OE-floor polling. This is because of

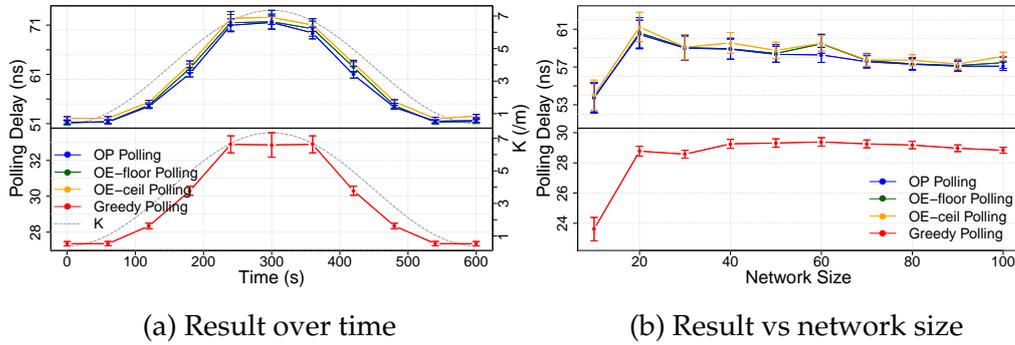


Figure 4.17: Polling delay for LSV channel

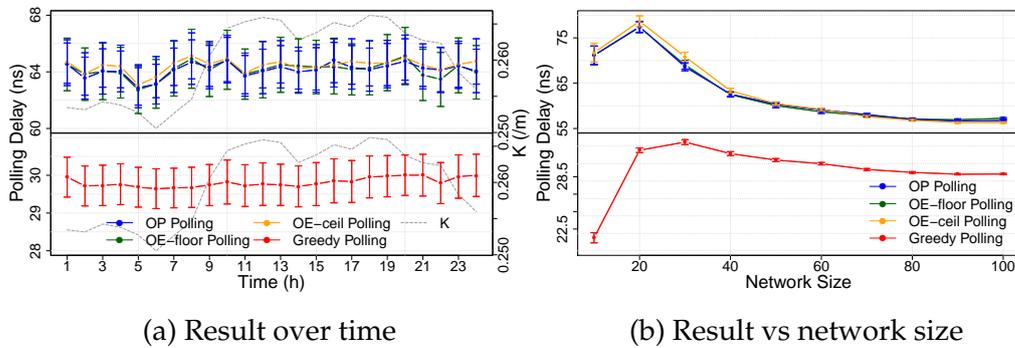


Figure 4.18: Polling delay for SSV channel

the packet aggregation size of 1, which happens when  $n = 0$  in line 6 of Algorithm 1, introduces a short transmission delay to the statistics of polling delay. OE-ceil polling has higher delay than OE-floor polling because of the potential of polling one more packet per sink which increases transmission delay. When the number of nano-sinks connected to the IoT gateway increases, as shown in Fig. 4.17b and 300s to 600s in Fig. 4.17a, the delay difference between OP polling / OE polling and greedy polling becomes small because of the decreased packet aggregation size with short packet transmission delay caused by the increased network size. It is envisaged that the delay performance of OP polling and OE polling would be better for EM-WNSNs with large packet aggregation size. For LSV channel,

delay shows significant relationship with the end-to-end hop count determined by connectivity whereas the mean values of delay over time for SSV channel are not significantly different from one to another due to the small impact of  $K$ .

**Collision Probability**

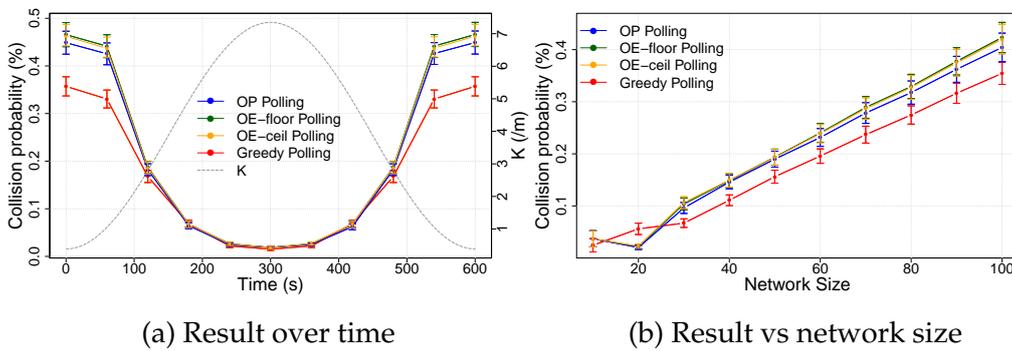


Figure 4.19: Collision probability for LSV channel

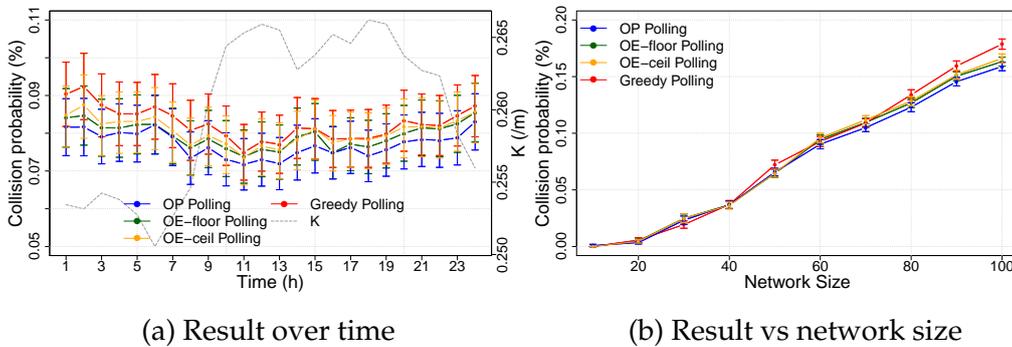


Figure 4.20: Collision probability for SSV channel

Overall, collision is related to network density determined by  $K$  and network size. For LSV channel shown in Fig. 4.19a and Fig. 4.19b, OP polling and OE polling show slight higher collision probability than greedy

polling due to two reasons: 1) multiple handshakes for information collection introduce more short beacons into the network than greedy polling; 2) shortened packet aggregation size reduces waiting time for medium access and therefore message packets are more likely to collide with beacons traversing in the network.

The second reason also leads to the slight difference between OE-floor polling and OE-ceil polling. Compared with OE polling, fewer collisions happen in OP polling because of the heterogeneous sizes of aggregated packets given by the probabilistic process, which avoids the collision that is supposed to happen in OE polling. Fortunately, the TS-OOK modulation used in EM-WNSNs significantly lowers the collision probability. The difference of collision probability between OP polling and greedy polling is less than 0.1%. For SSV channel in Fig. 4.20a and Fig. 4.20b, greedy polling and OE-ceil polling exhibit higher collision probability which resembles the result at 420s or network size is 20 in Fig. 4.19a and Fig. 4.19b due to the poor connectivity.

## Fairness

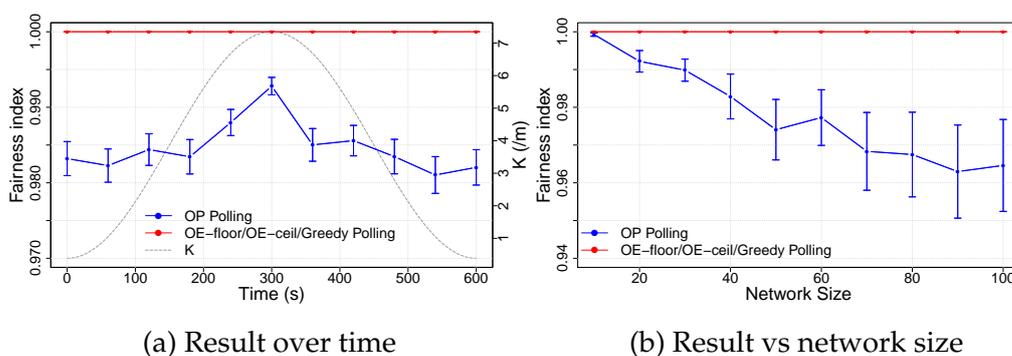


Figure 4.21: Fairness index for LSV channel

The fairness versus time and network size are shown in Fig. 4.21a - 4.22b, respectively. Unfairness is the result of the heterogeneous packet

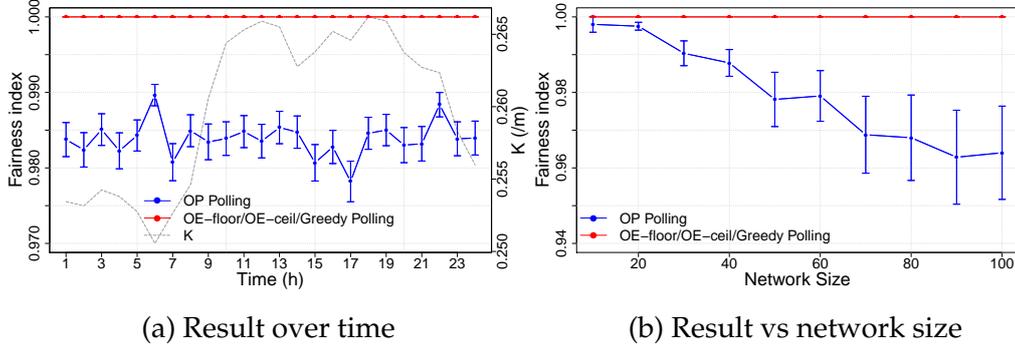


Figure 4.22: Fairness index for SSV channel

aggregation size which causes heterogeneous throughput of each sink. Therefore, benchmark schemes with homogeneous packet aggregation size consistently show a fairness index of 1. In comparison, with the probabilistic process that evenly distributes residual bandwidth to nano-sinks, OP polling shows an averaged fairness index of 0.9836. For OP polling, fairness is less affected when the percentile difference of throughput among sinks, which is either  $\frac{1}{n}$  or  $\frac{1}{n+1}$ , is small. Therefore, the fairness index rises for an increased  $\alpha$  or a decreased  $N_{max}$  (caused by an increased  $K$  or a decreased network size) that increases  $n$  as shown in line 5 of Algorithm 1, and vice versa.

Table 4.3: LSV-channel performance summary

Metric	OP polling	OE-floor polling	OE-ceil polling	greedy polling
$BE$ (%)	98.8719	80.8269	100	100
$P_{drop}$ (%)	0.4347	0	23.4069	78.2797
$E$ (pJ)	1758.4040	1444.5571	2105.5183	5678.6661
$L$ (ns)	58.3989	58.6974	59.2830	29.0738
$C$ (%)	64.7642	81.6603	91.0117	91.5128
$P_{COLL}$ (%)	0.2091	0.2171	0.2160	0.1761
$F$	0.9846	1	1	1

Table 4.4: SSV-channel performance summary

Metric	OP polling	OE-floor polling	OE-ceil polling	greedy polling
$BE$ (%)	99.0112	80.2271	100	100
$P_{drop}$ (%)	0.4718	0	23.8003	80.0658
$E$ (f)	17.3899	14.7892	20.2223	59.7585
$L$ (ns)	63.6048	64.5308	63.8886	29.6624
$C$ (%)	77.4121	64.5929	90.6478	90.1462
$P_{COLL}$ (%)	0.0766	0.0789	0.0799	0.0821
$F$	0.9826	1	1	1

In summary, compared with benchmark schemes, OP polling achieves the best balance between IoT bandwidth efficiency and EM-WNSNs energy efficiency.

## 4.4 Summary

For EM-WNSNs connected to the IoT, data acquisition is challenged by the dynamic IoT bandwidth and THz channel conditions. These two constraints reduce the resource utilization efficiency as a result of the mismatch between the EM-WNSNs' demand and the IoT bandwidth. To avoid unnecessary resource consumption, On-demand Probabilistic (OP) polling is designed. By dynamically adjusting the packet aggregation process, OP polling tunes the data volume extracted from EM-WNSNs to match the IoT bandwidth. Compared with benchmarks, OP polling efficiently utilizes the given bandwidth through a probabilistic process. From performance evaluation, OP polling achieves both high bandwidth efficiency of the IoT link and high energy efficiency of EM-WNSNs.

## Chapter 5

# Forwarding for Polling-based IoNT

The polling-based multi-hop nanonetworks discussed in Chapter 4 are subjected to dynamic network topology. Therefore, a data dissemination scheme with channel adaptability is demanded. However, graphene-based nano-devices have high sensitivity but low compute, power and memory capacity [23], which limits the ability to adopt precise channel-aware techniques used in traditional macro-scale devices [147] and demands networking schemes with low implementation complexity. Consequently, existing works done for both macro-scale sensor networks and EM-WNSNs are hardly adopted as solutions due to either the high complexity or the lack of dynamic channel adaptivity.

A lightweight data dissemination scheme, which is the Time-To-Live (TTL)-based Efficient Forwarding (TEForward), is designed to achieve high energy efficiency for backhaul tier of polling-based EM-WNSNs under dynamic channel states. In this scenario, the IoT gateway broadcasts polling beacons periodically to poll nano-sinks for data extraction. In TEForward, the polling beacons have multiple functions including data collection and topology discovery. In each polling interval, nano-sinks extract the duplicate count and TTL values to infer the latest network topology information

to narrow the set of forwarders and direct data flows. TEFoward achieves forwarding with high data delivery ratio and energy efficiency under dynamic channel states with repurposing of polling beacons.

In this chapter, the optimal forwarding is firstly defined, followed by a description of the design of TEFoward. Finally, the performance of TEFoward is evaluated for both the single-gateway network and multi-gateway network.

## 5.1 Optimal Energy-efficient Forwarding

For EM-WNSNs with coordinated IoT gateways that collect data via polling under dynamic channel conditions, the energy consumption of nano-sinks is minimized when the message packets polled are directed to the gateways along the most energy-efficient path in the current topology. Therefore, the goal of optimal forwarding is to find the optimal path  $\rho^{K*}$  with the minimized end-to-end energy consumption. Supposing that unicast forwarding is adopted, the path selection of optimal forwarding is defined as below:

$$\text{Obj. } \rho^{K*} = \underset{\rho^K \in P^K}{\text{argmin}} E_{e2e}(\rho^K) \quad (5.1)$$

$$\text{where } E_{e2e}(\rho^K) = E_{e2e}^{TX}(\rho^K) + E_{e2e}^{RX}(\rho^K) \quad (5.2)$$

$$E_{e2e}^{TX}(\rho^K) = E_{TX} H_{e2e}^{\rho^K} \quad (5.3)$$

$$E_{e2e}^{RX}(\rho^K) = E_{RX} N_{RX-e2e}^{\rho^K} \quad (5.4)$$

$$E_{TX} = \gamma E_{RX} \quad (5.5)$$

where  $E_{e2e}$  is the end-to-end energy consumption composed of the end-to-end energy consumption for transmitting  $E_{e2e}^{TX}$  and receiving  $E_{e2e}^{RX}$ ,  $\rho^K$  and  $P^K$  denote the selected path and the path set under channel condition  $K$ ,  $E_{TX}$  and  $E_{RX}$  are the energy consumption for transmitting and receiving one message packet,  $\gamma$  denotes the numerical relationship between  $E_{TX}$  and  $E_{RX}$  that is assumed to be 10 [24],  $H_{e2e}^{\rho^K}$  is the end-to-end hop count for

path  $\rho^K$ ,  $N_{RX-e2e}^{\rho^K}$  is the total number of sinks receiving the transmission along path  $\rho^K$ .

In order to find the optimal path  $\rho^{K*}$ , each sink firstly builds a graph based on the coordinates and transmission range of sinks and gateways under the up-to-date  $K$ . In this graph, the length of an edge is the total energy consumption for forwarding to the corresponding sink, given by  $E_{TX} + N_D E_{RX}$  where  $N_D$  is the neighbour degree of the candidate forwarder. Next,  $\rho^{K*}$  of each sink is obtained by calculating the shortest paths to gateways via Dijkstra's algorithm and the most energy-efficient one is selected. Overall, the complexity is  $\mathcal{O}(N_{total}^2)$  where  $N_{total}$  is the total number of sinks and gateways. The process above demands high physical capacity with respect to energy, storage, and computation, which is not feasible for nano-devices. Therefore, the sub-optimal TEFoward with low resource demands is designed as an alternative.

## 5.2 TTL-based Efficient Forwarding

In this section, the design of TEFoward is discussed. The pseudocode of TEFoward is shown in Algorithm 6, Fig. 5.2, and Fig. 5.3 with notations explained in Table 5.1.

### Overview and Assumptions

The key concept of TEFoward is to leverage the polling beacons to obtain the latest network topology information under dynamic channel states for forwarder selection and data diffusion. As shown in Fig. 5.2 (a) and Fig. 5.3 (a), for each polling, the IoT gateway floods a polling *Beacon* that has packet format as shown in Fig. 5.1 with TTL value of  $TTL_{max}$  for data collection. Next, following Fig. 5.2 (b) and Fig. 5.3 (b), nano-sinks select forwarders based on the path information extracted from *Beacon* and respond the gateway with message packets.

The TEFoward mechanism relies on the following assumptions to operate as designed:

1. The channel condition remains constant during one polling process which starts from beacon dissemination and finishes with data collection at the gateway. This assumption is consistent with the assumption in Section 4.2.1.
2. In the multiple-gateway scenario, IoT gateways are coordinated for synchronous beacon dissemination and aware of each other's coordinates.
3. Besides the communication modules for connecting macro-scale devices, IoT gateways are also equipped with the same transceivers as

Table 5.1: Notations used to describe TEFoward

Notation	Description
$Beacon$	A beacon packet
$Beacon_{DUP}^i$	The $i$ -th beacon duplicate received by a sink
$M_{pkt}$	An aggregated message packet
$TTL_S^j$	The TTL setting of a nano-sink $j$
$TTL_P$	The TTL value of a packet
$TTL_{max}$	The maximum value of $TTL$
$N_S^j$	The neighbour-size indicator of a nano-sink $j$
$CumN_S^j$	The indicator of the cumulative neighbour-size of the selected forwarder of sink $j$
$CumN_P$	The indicator of the cumulative neighbour-size field of a beacon sender
$T_{wait}$	The waiting time for beacon collection
$\gamma$	The ratio of $E_{TX}$ to $E_{RX}$
$FID_S^j$	The ID of the selected forwarder of sink $j$
$SID_P$	The ID of a beacon sender



Figure 5.1: Structure of a polling beacon.

nano-sinks to enable THz communications.

4. Nano-sinks are active during one polling interval for beacon and message dissemination.

### Polling Beacon Structure

Besides the original packet header [145], the polling beacon in TEFoward adopts two new fields that are  $CumN_P$  and  $T_{wait}$  to assist forwarder selection.

Specifically,  $CumN_P$  is an indicator for inferring the cumulative receiver count that implies the total number of nano-sinks receiving message packets along the end-to-end path. This parameter indicates the end-to-end energy consumption caused by the packet reception process.  $CumN_P$  is set to 0 by the IoT gateway and updated by nano-sinks to cumulate the local neighbour size indicators during the beacon flooding.

$T_{wait}$  specifies a time window for collecting beacon duplicates. To comprehensively collect beacon duplicates from neighbours,  $T_{wait}$  is conservatively set to  $\frac{d_S}{c}$  whereby  $d_S$  is the transmission range of nano-sinks and  $c$  is the speed of light. Considering the computation capacity of nano-devices, the calculation of  $T_{wait}$  is delegated to the IoT gateways. A heuristic for calculating  $T_{wait}$  is presented as follows. Before each polling, the IoT gateways coordinate with each other in THz band to derive the up-to-date  $K(f)$  for the current environment by solving the equation below:

$$A_T(f, d) = P_{TX} - S_{RX}, \quad (5.6)$$

where  $P_{TX}$  and  $S_{RX}$  are the transmission power and receiving sensitivity. This operation is feasible since all the variables could be easily obtained

for gateways. Next, with  $K(f)$ , the transmission power and receiving sensitivity of nano-sinks as inputs, the critical transmission range  $d_S$  of nano-sinks could be derived by solving Eqn. (5.6).

Finally,  $T_{wait}$  is calculated based on  $d$  as below:

$$T_{wait} = \begin{cases} 0, & d_S \geq d_{max} \\ \frac{d_S}{c}, & d_S < d_{max}, \end{cases} \quad (5.7)$$

where  $d_{max}$  denotes the network diameter that is the maximum distance between two network devices in the network. For a single-hop network, nano-sinks respond to polling beacons directly without invoking the forwarder selection algorithm.

If there is only one IoT gateway, the calculation of  $T_{wait}$  requires external information or the communication between the gateway and nano-sinks.

### Forwarder Selection

IoT gateways initiate the data collection process by simultaneously broadcasting the same polling beacon. Each nano-sink  $j$  that receives a new *Beacon* first buffers it and aligns its TTL setting  $TTL_S^j$  based on the beacon  $TTL_P$  to obtain its distance to the IoT gateway in number of hops (cf: line 3 in Algorithm 6). Next, following line 4, nano-sinks reset the indicator of the local neighbour-size  $N_S^j$  and initialize the local variable  $CumN_S^j$  with  $CumN_P$  carried by *Beacon*. For each nano-sink,  $CumN_S^j$  is the  $CumN_P$  of the current forwarder selected. It plays a significant role in forwarder selection since it is used to identify the forwarder which directs packets along the path that triggers the fewest number of packet receptions. After the above settings, a timer that equals  $T_{wait}$  as specified in *Beacon* is set for collecting beacon duplicates, as shown by line 5.

In lines 6 - 17 of Algorithm 6, on receiving a beacon duplicate  $Beacon_{DUP}^i$ , the nano-sink  $j$  increments its  $N_S^j$  and updates its forwarder selection if the sender of  $Beacon_{DUP}^i$  establishes a path with higher energy efficiency than

---

**Algorithm 6** TTL-based Efficient Forwarding

---

**Initialization**

$$N_S^j = 0, CumN_S^j = 0, CumN_P = 0, TTL_S^j = TTL_{\max}, TTL_P = TTL_{\max}$$
**For each nano-sink  $j$ :**

- 1: **if** receives a new polling beacon *Beacon* **then**
  - 2:   Buffer *Beacon*
  - 3:    $TTL_S^j = TTL_{\max} - TTL_P$
  - 4:   Set  $N_S^j = 1, CumN_S^j = CumN_P, FID_S^j = SID_P$
  - 5:   Set timer  $T_S^j = T_{wait}$
  - 6:   **while**  $T_S^j \neq 0$  **do**
  - 7:     **for** the  $i$ -th *Beacon* duplicate  $Beacon_{DUP}^i$  **do**
  - 8:        $N_S^j = N_S^j + 1$
  - 9:       **if**  $CumN_S^j + \gamma TTL_S^j > CumN_P + \gamma TTL_P$  **then**
  - 10:           $CumN_S^j = CumN_P$
  - 11:           $FID_S^j = SID_P$
  - 12:          Clear beacon buffer and buffer  $Beacon_{DUP}^i$
  - 13:       **else**
  - 14:          Drop  $Beacon_{DUP}^i$
  - 15:       **end if**
  - 16:     **end for**
  - 17:   **end while**
  - 18:   Set  $CumN_P = CumN_S^j + N_S^j$
  - 19:   Broadcast the buffered beacon with  $TTL_P = TTL_P - 1$
  - 20:   Aggregate and send  $M_{pkt}$  to  $FID_S^j$
  - 21: **else if** receives a new message packet  $M_{pkt}$  **then**
  - 22:   **if** this packet is for me **then**
  - 23:     Forward  $M_{pkt}$  with  $TTL_P = TTL_P - 1$  to  $FID_S^j$
  - 24:   **end if**
  - 25: **end if**
-

the previous record, via line 9 that compares the end-to-end energy consumption in Eqn. (5.2) of a potential forwarder and that of the current selection. If higher energy efficiency is achieved, the nano-sink selects the sender of this duplicate as its forwarder by updating the forwarder ID  $FID_S^j$  with the sender's ID  $SID_P$  and replaces the buffered beacon with this duplicate; otherwise, the duplicate beacon is dropped.

In line 18, when the timer  $T_S^j$  counts to zero,  $CumN_P$  of the buffered beacon is updated by summing  $CumN_S^j$  and  $N_S^j$  to add the impact of current sink into the path traversed by the beacon duplicate. Next, the buffered beacon is re-broadcasted for assisting other sinks with forwarder selection. Ultimately, the nano-sink aggregates packets from nanosensors and sends the aggregated packet  $M_{pkt}$  to the selected forwarder  $FID_S^j$ . Sinks that are involved in packet forwarding will forward this message packet to their own selected forwarder as in lines 21–25 until it reaches the IoT gateway. In this way, all message packets are effectively and efficiently directed to one of the IoT gateways along the latest efficient path of the polling cycle.

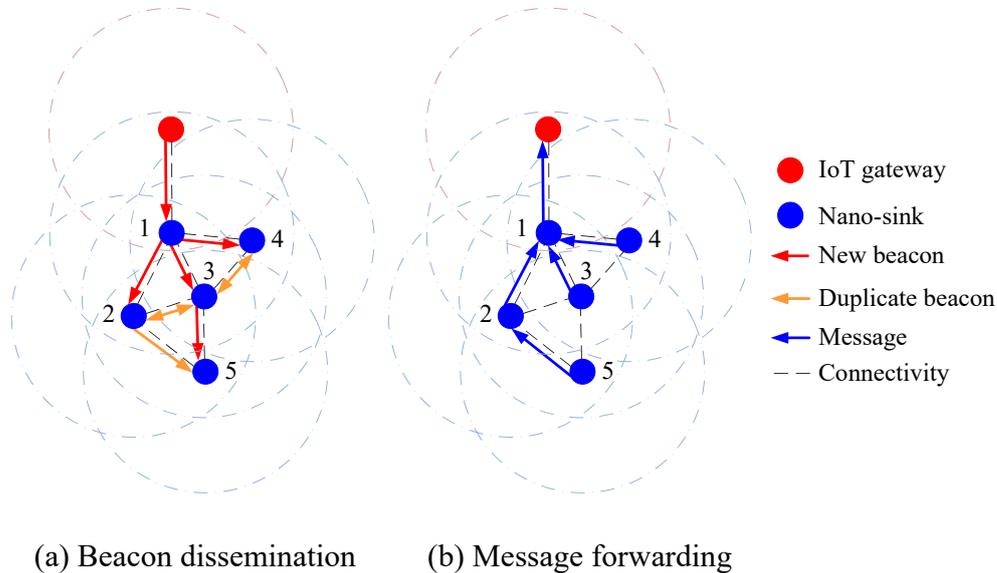


Figure 5.2: TEFoward example for a single-gateway network.

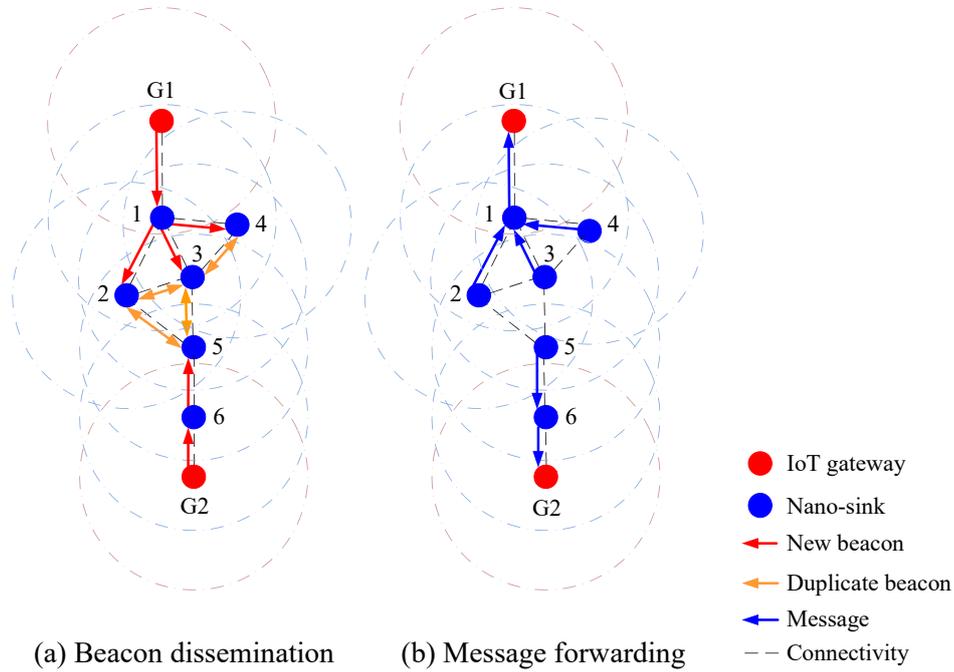


Figure 5.3: TEFoward example for a multi-gateway network.

The example paths selected by TEFoward for the single-gateway network are shown in Fig. 5.2(b). “sink 5” eventually selects “sink 2” as its forwarder because it involves the least number of transmitting and receiving operations. Similarly for the multi-gateway network as depicted in Fig. 5.3(b), the beacon from “gateway 1” is not forwarded by “sink 5” for whom “gateway 2” is more energy-efficient to reach. The final forwarder selected will direct packets towards the IoT gateway that incurs minimal energy costs to reach. As a result, the network is partitioned by multiple gateways with high energy efficiency for nano-sinks, as shown in Fig. 5.3(b) whereby “sink 1 - 4” would forward data towards “gateway 1” whereas “sink 5” and “sink 6” would deliver data to “gateway 2”.

For each nano-sink, the TEFoward algorithm shows linear complexity of  $\mathcal{O}(N_S^j)$  as its forwarder selection is based on the information obtained from  $N_S^j$  neighbours. Compared with the existing solutions, the merits

of TEFoward are: 1) feasible implementation for nano-devices as TEFoward implements major networking functions assisted by polling beacons, thus fits the computation capacity of nano-devices with  $\mu\text{m}$ -level dimensions [6]; and 2) requiring no hardware support beyond nano-devices such as RSSI module and GPS. TEFoward provides the first solution for data dissemination in nanoscale sensor networks that considers dynamic environment impact for the backhaul tier of polling-based EM-WNSNs with coordinated IoT gateways.

## 5.3 Performance Evaluation

In this section, the performance of TEFoward is evaluated and analysed against benchmark schemes.

### 5.3.1 Performance Metrics

In this section, TEFoward is evaluated based on the following performance metrics:

1. Packet delivery ratio,  $PDR$ :

$$PDR = \frac{N_R}{N_T}, \quad (5.8)$$

where  $N_R$  is the total number of unique packets received by the IoT gateway, and  $N_T$  is the total number of unique aggregated message packets transmitted by nano-sinks for one polling. Packet delivery ratio measures the impact of path selection on end-to-end packet delivery.

2. Collision probability,  $P_{COLL}$  as defined by Eqn. 4.12 in Section 4.3.1:

$$P_{COLL} = \frac{N_{COLL}}{N_{RX}}, \quad (5.9)$$

where  $N_{COLL}$  is the total number of collision and  $N_{RX}$  is the total number of packet receptions during the end-to-end data delivery for one polling interval. For most of the forwarding schemes, collision is the main reason for packet loss.

3. Delay,  $D$ :

$$D = T_{RX} - T_{TX}, \quad (5.10)$$

where  $T_{RX}$  is the time when a packet is received by the IoT gateway, and  $T_{TX}$  is the time when the polling beacon is transmitted by the gateway. This metric measures the impact of forwarding on polling latency.

4. Energy consumption of polling one packet,  $E$ :

$$E = E_T + E_R, \quad (5.11)$$

where  $E_T$  and  $E_R$  are the total energy consumed by transmitting and receiving packets during data delivery for polling one packet, respectively. Energy consumption is the key metric for TEFoward.

5. Forwarder degree,  $FD$ : the number of forwarders for each transmission during data delivery

$$FD = \frac{N_F}{H}, \quad (5.12)$$

where  $N_F$  is the total number of physical transmission and  $H$  is the total hop count of the packets received by the gateway. Forwarding degree measures the redundancy of forwarding during end-to-end delivery. For optimal energy efficiency,  $FD = 1$  since only one forwarding occurs for each hop.

6. Cumulative receiver count,  $CR_{count}$ : the total number of nano-sinks involved in packet reception during the end-to-end data delivery for one message packet polled

$$CR_{count} = \frac{N_{RX}}{N_R}, \quad (5.13)$$

where  $N_{RX}$  is the number of packet receptions occurring during the end-to-end data delivery for one polling. This metric is used for inferring the end-to-end energy consumption for receiving packets.

7. Fairness,  $F$ :

$$F = \frac{(\sum_{i=1}^{N_{sink}} \lambda_i)^2}{N_{sink} \sum_{i=1}^{N_{sink}} \lambda_i^2}, \quad (5.14)$$

where  $N_{sink}$  is the number of sinks polled and  $\lambda_i$  is the throughput of packets transmitted by the  $i$ -th sink during one polling interval.

### 5.3.2 Benchmarks

To benchmark the performance of TEFoward, the following benchmarks are adopted:

1. The optimal forwarding (opt).
2. The TTL-based Efficient Flooding (TEFlood) presented in Algorithm 5. In TEFlood, nano-sinks align the TTL settings of their message packets based on polling beacons and forward message packets only when they are located closer to the IoT gateway than the sender to minimize the flooding area.
3. The Selective flooding (S-Flooding) [145].
4. The Random forwarding (RForward) [145].

Similar to TEFoward, a nano-sink in RForward waits for  $T_{wait}$  upon receiving a unique polling beacon to collect neighbour information and then randomly selects one neighbour as its forwarder. In S-Flooding and RForward, if a nano-sink receives a polling beacon directly from the gateway, the message would be directly transmitted to the gateway by specifying a specific MAC flag. Besides, each sink only forwards a packet once.

### 5.3.3 Simulation Setup

The performance of TEFoward is evaluated using the nano-sim [145] package of NS3. Simulation settings are shown in Table 5.2. The 100-fs-long Gaussian pulse with 1 pJ energy [24] on 1 THz is adopted for one bit transmission. A message packet is 136 bits long composed of 100-bit payload and 36-bit overhead with an initial TTL of 100 [145]. Statistically, it is assumed that the average ratio of bit “1” in a packet is 50% [24] when evaluating energy consumption. Considering the property of graphene [6], the receiving sensitivity for packet reception is set by expectation as researchers do in a recent work [129].

The simulated area is set to  $10\text{m} \times 10\text{m}$  wherein 10 to 100 static nano-sinks are uniformly distributed so that the network connectivity transits from low to high. For the single-gateway network, one IoT gateway is placed in the centre while for the multi-gateway scenario three gateways are uniformly randomly deployed following a uniform distribution with a mean of 5m for both coordinates.

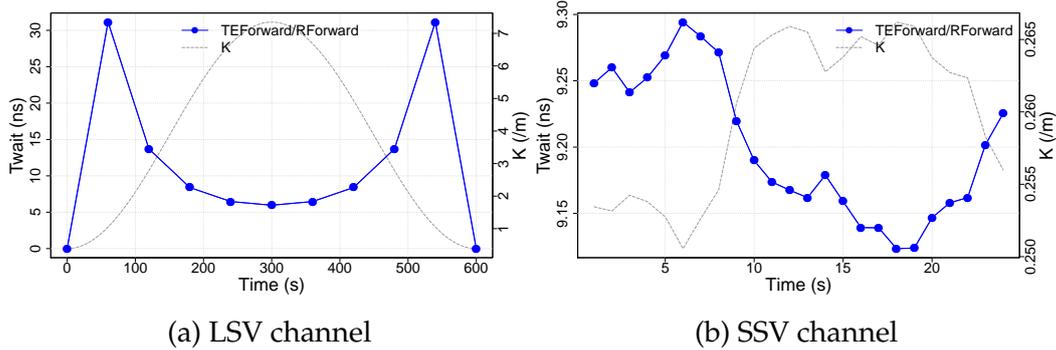
The corresponding size of field  $CumN_P$  is set to 10 bits. Considering the energy capacity of nano nodes, a nano-sink aggregates 10 packets for each polling interval. Two channel scenarios, which are LSV channel and SSV channel mentioned in Chapter 4, are adopted.

The  $K$  and the corresponding  $T_{wait}$  over simulation time for LSV channel and SSV channel are shown in Fig. 5.4a and Fig. 5.4b, respectively. It is worth noting that for LSV channel,  $T_{wait} = 0$  at 0s and 600s because the network is one-hop connected.

Performance of TEFoward is evaluated and compared with optimal forwarding, TEFlood, RForward, and S-Flooding via the metrics in Section 5.3.1. Each result presented is the average with 95% confidence interval obtained from 50 simulations with different seed values.

Table 5.2: Simulation parameters

Parameter	Value
Frequency	1 THz
Pulse energy	1 pJ / 10 aJ
Pulse duration	100 fs
Pulse interval	10000 fs
Receiving sensitivity	-130 dBm
$Cum.N_P$ field size	10 bits
$T_{wait}$ field size	15 bits
Unit packet size	136 bits
Beacon packet size	46 bits
Packet aggregation size	10 packets
$TTL_{max}$	100
Simulation area	10m $\times$ 10m
Network size	10,20,30,40,50,60,70,80,90,100 nano-sinks
IoT gateway position	Centre / uniformly random
Nano-sink position	Uniformly distributed
Nano-sink mobility	Static
Simulation duration	600 s / 24 h
Polling interval	60 s / 1 h

Figure 5.4:  $T_{wait}$  vs time

### 5.3.4 Results and Analysis

The simulation results for different topologies and channel conditions are presented in Fig. 5.5a - 5.28b. For clear observation, graphs are split vertically when the results for different schemes present huge differences in Y axis. The overall performance evaluation for different scenarios is summarized in Table 5.3 - 5.6.

#### Packet Delivery Ratio

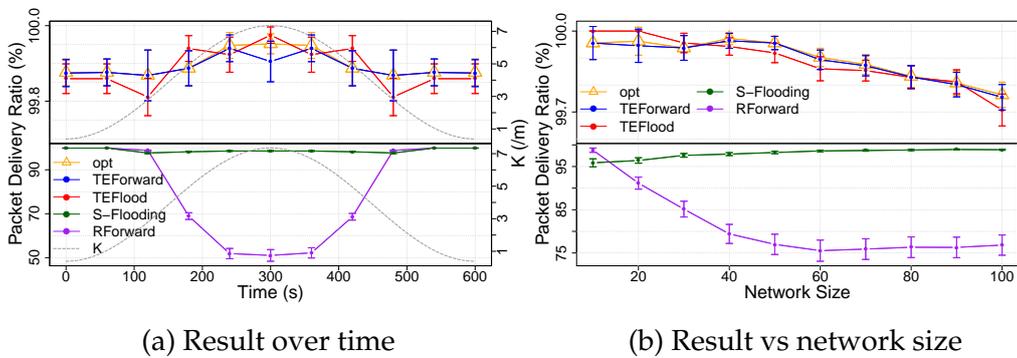


Figure 5.5: Packet delivery ratio for SGW with LSV channel

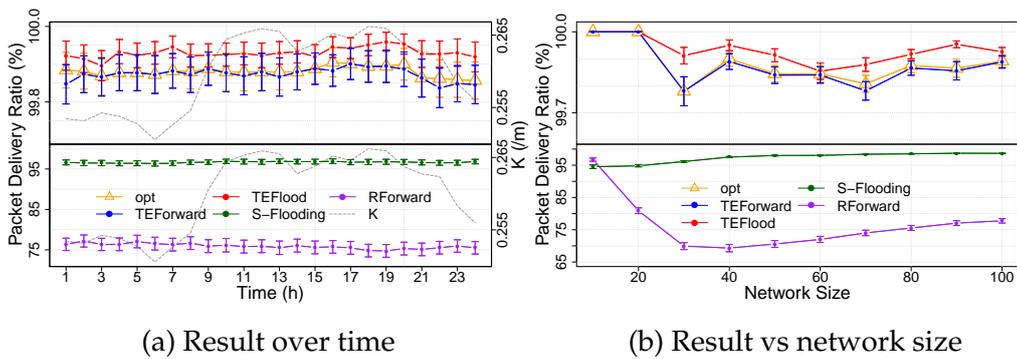


Figure 5.6: Packet delivery ratio for SGW with SSV channel

As shown in Fig. 5.5a - 5.8b, benefiting from the Time Spread On-Off Keying (TS-OOK) [14] modulation with low collision probability, all

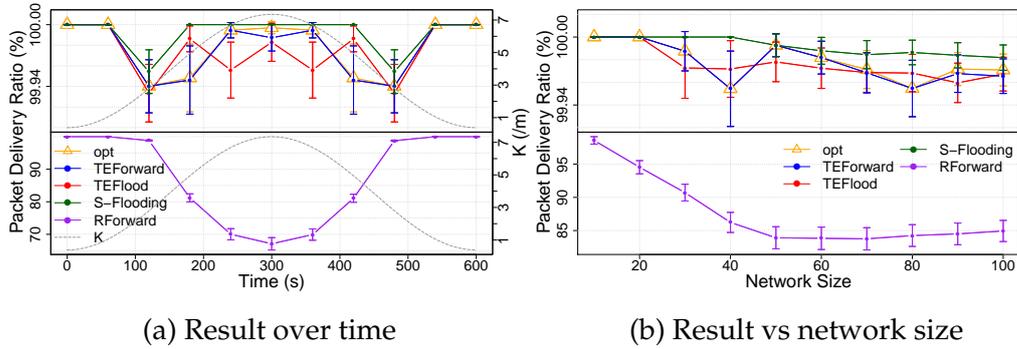


Figure 5.7: Packet delivery ratio for MGW with LSV channel

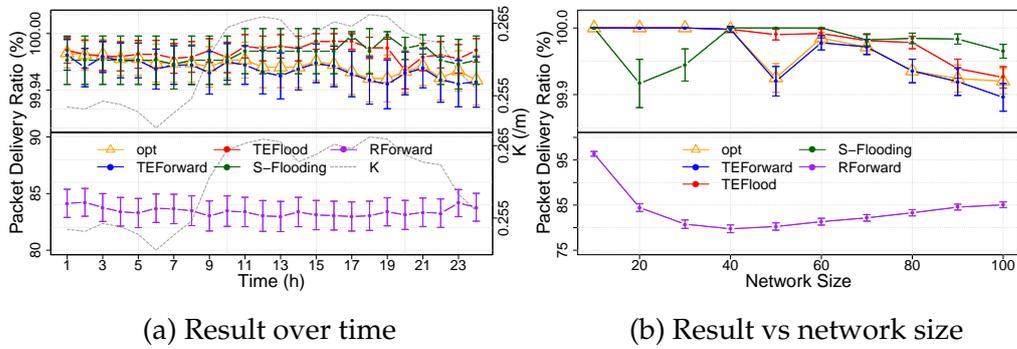


Figure 5.8: Packet delivery ratio for MGW with SSV channel

schemes except RForward show high packet delivery ratio that is above 99% on average.

Packet delivery ratio deteriorates due to two reasons: collision and forwarding failure. The packet delivery ratio of optimal forwarding, TEForward, and TEFlood is affected by collisions at intermediate hops that are positively related to neighbour density. Generally, packet delivery ratio is high for large  $K$  and small network sizes that reduce node density. One exception is seen at 0s, 60s, 540s, and 600s because the transmissions are mostly single-hop under such channel conditions and therefore robust against collisions at sinks.

In contrast, RForward additionally suffers from forwarding failure due

to the potential reverse path that directs a packet back to the sender or the intermediate forwarder that refuses to forward this packet again. Generally, RForward exhibits a higher probability of forwarding failures for low neighbour density or more end-to-end transmissions happen since the forwarder at each hop is randomly selected. Therefore, RForward shows a decreased packet delivery ratio for increased  $K$  which results in more hops and forwarding candidates. However, the impact of network size is not simply monotonic. On the one hand, an increased network size leads to more hops that deteriorates the end-to-end delivery. On the other hand, however, an increased network size mitigates forwarding failure because of the increased number of forwarding candidates that decrease the probability of forwarding one packet back to the previous forwarders, which eventually compensates packet delivery ratio.

Similar behaviours are observed for S-Flooding in single-gateway scenarios. Packet delivery ratio is low for small network sizes due to the high homogeneity of path selection that is vulnerable for collision. However it becomes high when network size increases because of the heterogeneous paths formed. Nevertheless, this phenomenon is not presented for multi-gateway scenarios as a result of the high heterogeneity of paths established by multiple gateways.

### **Collision Probability**

From Fig. 5.9a - 5.12b, all schemes show low collision probability that is below 0.3% attributed to the TS-OOK modulation. Generally, collision probability increases for a increased node density that is caused by a decreased  $K$  and increased network sizes. Therefore, optimal forwarding achieves the lowest collision probability whereas schemes with multi-cast feature like S-Flooding and TEFlood confront more collisions.

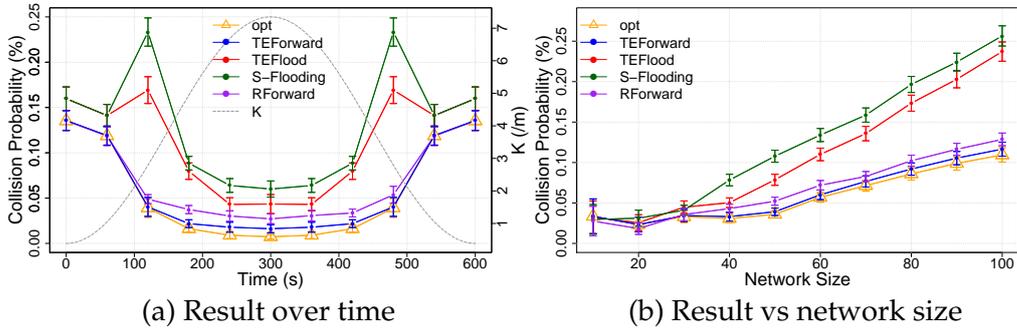


Figure 5.9: Collision probability for SGW with LSV channel

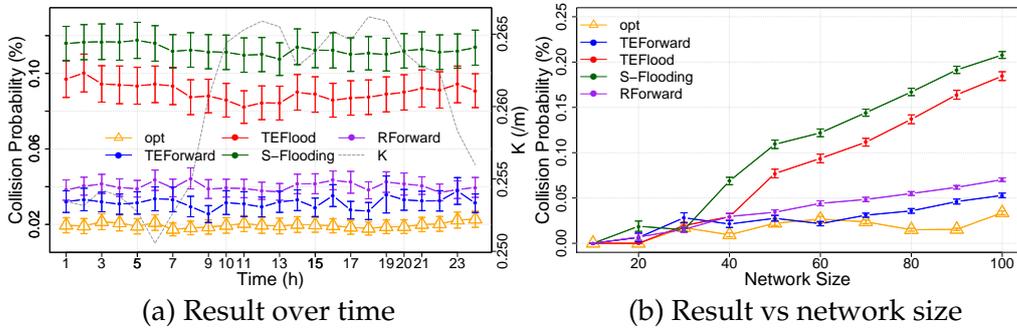


Figure 5.10: Collision probability for SGW with SSV channel

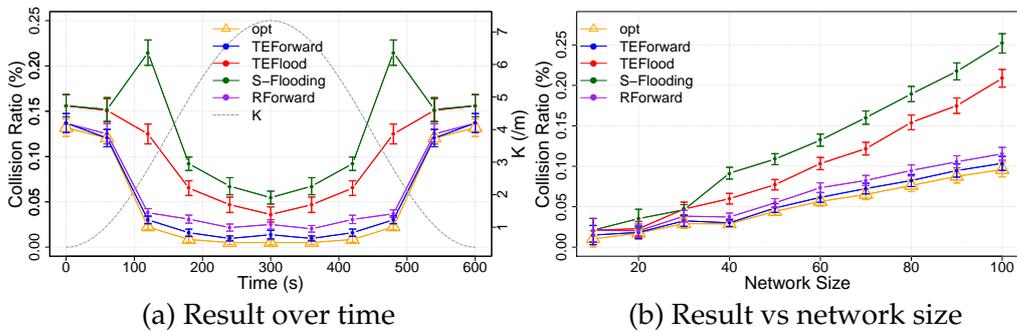


Figure 5.11: Collision probability for MGW with LSV channel

### Average Delay

Generally, delay is directly proportional to the hop count of the paths that packets traverse which are determined by network connectivity. The

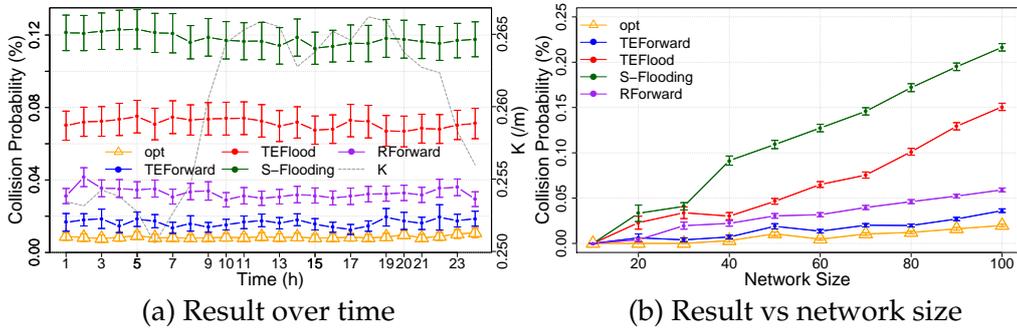


Figure 5.12: Collision probability for MGW with SSV channel

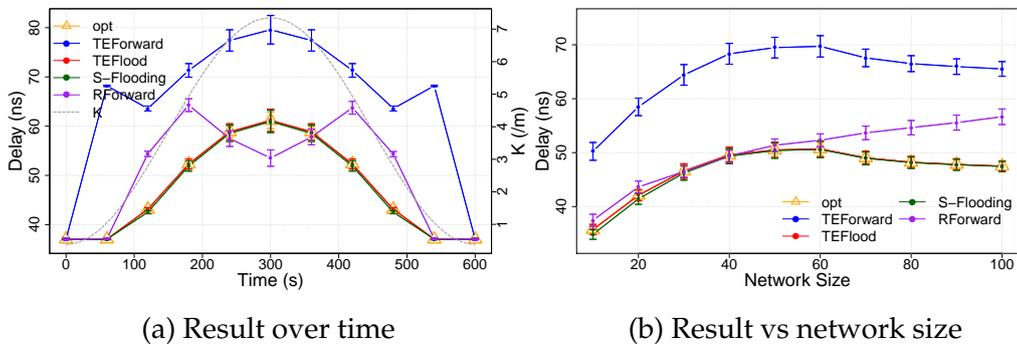


Figure 5.13: Delay for SGW with LSV channel

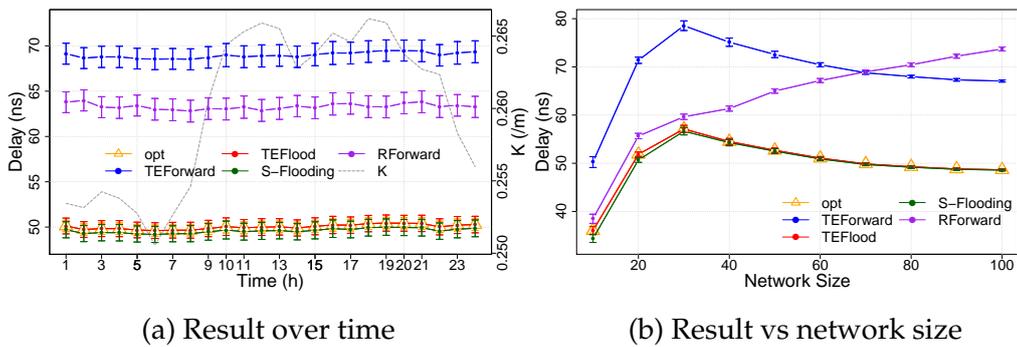


Figure 5.14: Delay for SGW with SSV channel

TEForward and RForward exhibit delay that is higher than other schemes due to the waiting time for beacon collection at each hop. Therefore the de-

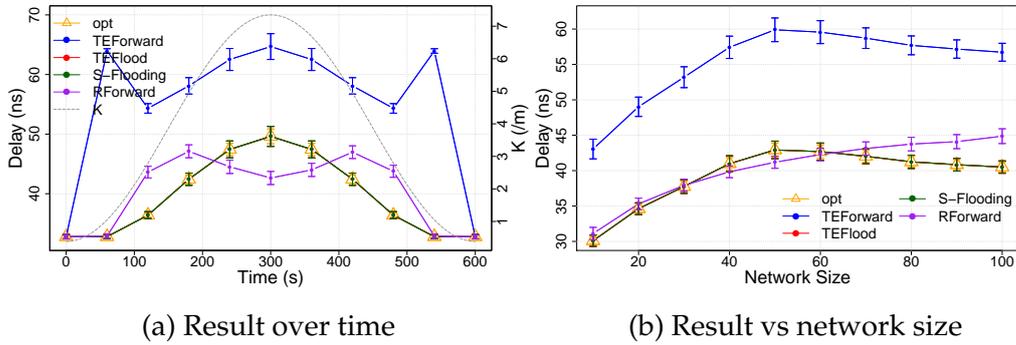


Figure 5.15: Delay for MGW with LSV channel

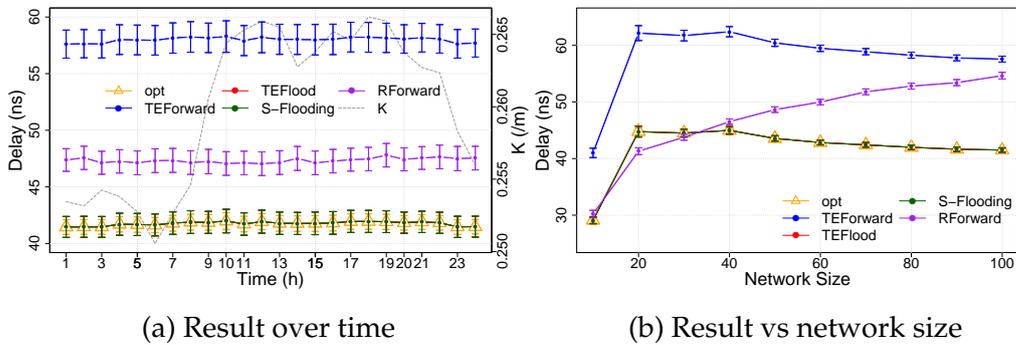


Figure 5.16: Delay for MGW with SSV channel

lay of TEForward is highly correlated with  $T_{wait}$  shown in Fig. 5.4a when the hop count is low at 0s, 60s, 540s, and 600s. However, TEForward's average delay is only 17.41 ns higher than that of S-Flooding with the lowest delay because of the small-scale coverage of nanonetworks. For LSV channel, delay increases as a result of the high molecular absorption  $K$  that leads to an increased hop count, and vice versa, as shown in Fig. 5.15b. RForward shows a different trend at 240s, 300s, and 360s due to the poor packet delivery ratio such that only packets from sinks close to the gateway are delivered and involved in delay statistics, which also results in the observation that differs from TEForward at 60s and 540s.

In Fig. 5.15a, for all schemes except RForward, delay firstly increases as

a result of improved connectivity which allows sinks multi-hop away from the IoT gateway to be polled. When network size further increases, the chance of having forwarders geographically close to a gateway at each hop increases, which then decreases the end-to-end delay. However, it requires higher network density for the delay of RForward to show the same trend due to its lower probability for delivering packets than other methods. For SSV channel, RForward shows higher delay which eventually exceeds that of TEFoward as demonstrated in Fig. 5.16a and Fig. 5.16a since the end-to-end hop count of SSV channel is on average higher than that of LSV channel.

### Total Energy Consumption

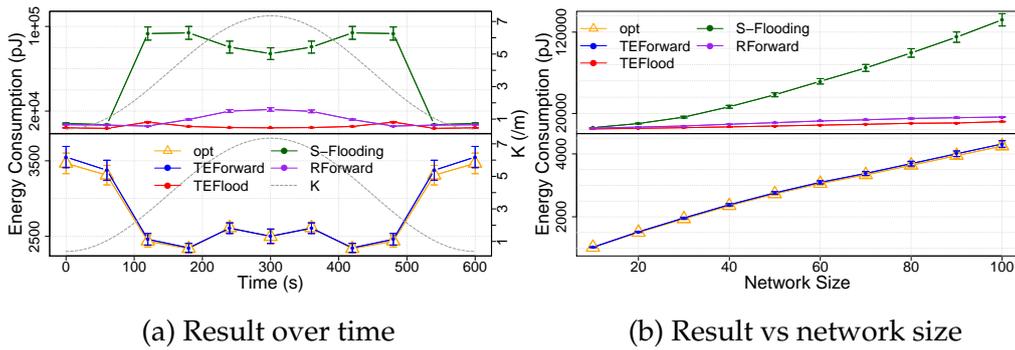


Figure 5.17: Energy consumption for SGW with LSV channel

From Fig. 5.17a - 5.20b, TEFoward shows low energy consumption that is on average only 35.75 pJ and 0.0685 fJ higher than optimal forwarding for LSV channel and SSV channel, respectively. This difference is caused by two factors: first, although a conservative  $T_{wait}$  is adopted at each hop, it is still possible for a nano-sink to miss the beacon duplicate that comes from a path with lower energy consumption; second, the additional fields  $CumN_P$  and  $T_{wait}$  intuitively increase the energy consumption during beacon dissemination. Therefore, the difference and the absolute value of en-

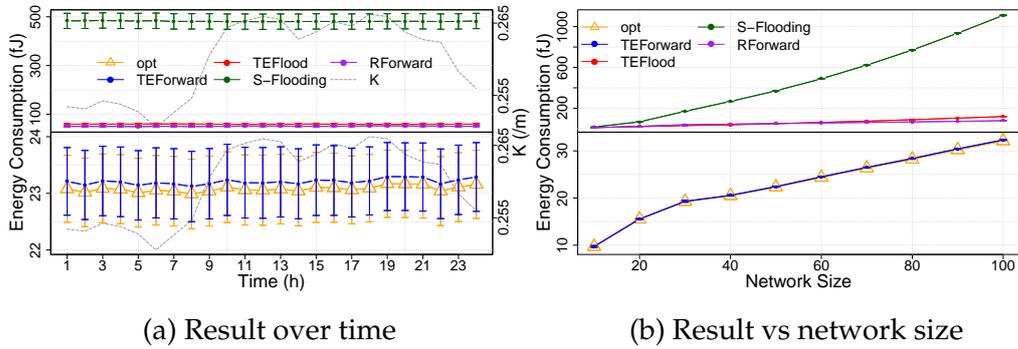


Figure 5.18: Energy consumption for SGW with SSV channel

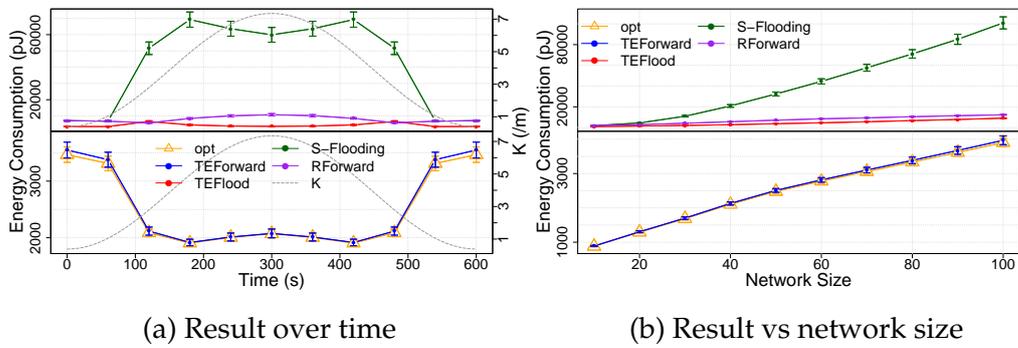


Figure 5.19: Energy consumption for MGW with LSV channel

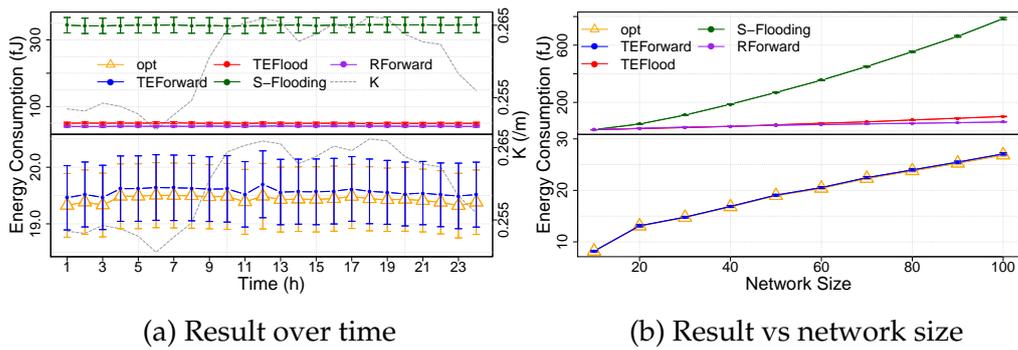


Figure 5.20: Energy consumption for MGW with SSV channel

energy consumption between TEForward and optimal forwarding are large for small  $K$  and large network sizes that cause high node density which triggers a large number of transmissions and receptions during beacon flooding. From the results of LSV channel, S-flooding and RForward show low energy consumption when  $K$  is low since the proportion of one-hop communications is high and sinks are enhanced to directly transmit towards the gateway. However, RForward suffers from forwarding failure that cause unnecessary energy for high  $K$ . The TEForward presents the similar behaviour as S-Flooding but with lower energy consumption because flooding area is constrained. For SSV channel, the results over time do not exhibit significant change due to the small percentile of molecular absorption loss in the total path loss.

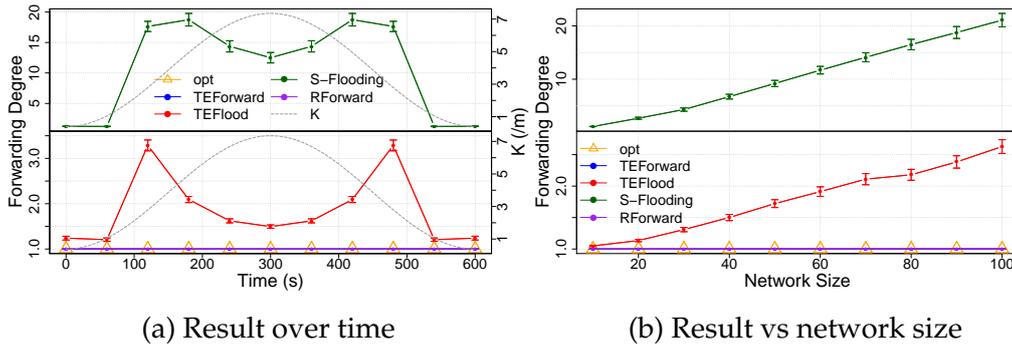


Figure 5.21: Forwarder count for SGW with LSV channel

For all schemes except RForward in LSV channel, there exists a strong inference from forwarding degree and cumulative receiver count shown in Fig. 5.21a - 5.28b, respectively. It can be seen that the difference between TEForward and optimal forwarding is tiny regarding the cumulative receiver count because of the high similarity of path selection. Nevertheless, RForward is an exception since the statistics of these two metrics are extracted from the packets successfully delivered, which leads to biased results considering the high packet loss ratio of RForward. It is worth noticing that the multicast feature of TEFlooding and S-Flooding re-

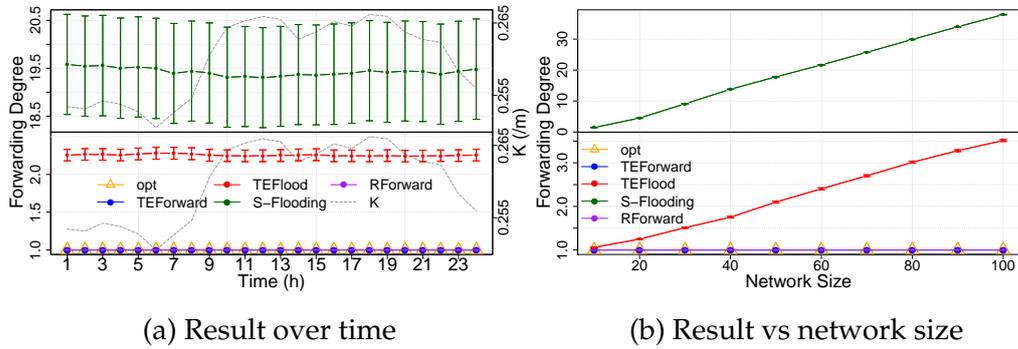


Figure 5.22: Forwarder count for SGW with SSV channel

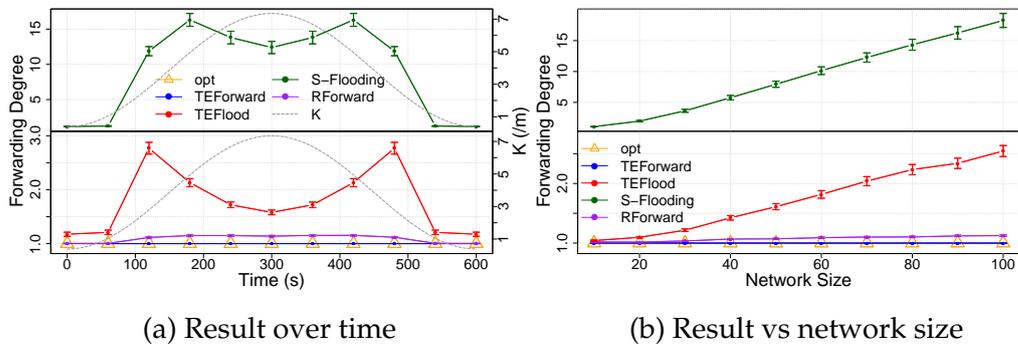


Figure 5.23: Forwarder count for MGW with LSV channel

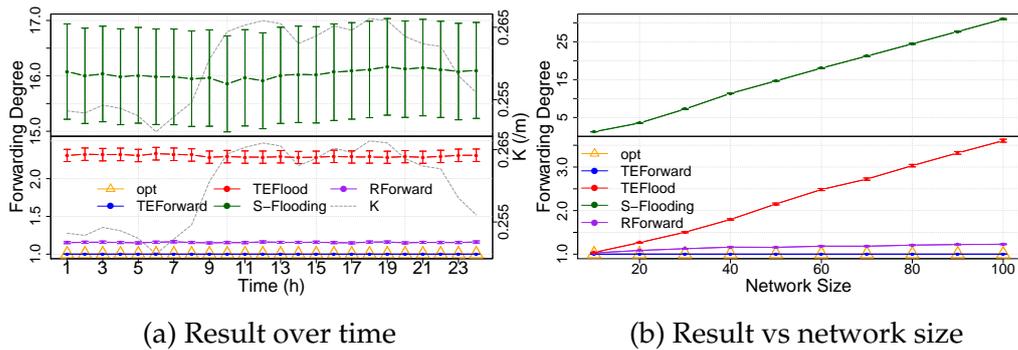


Figure 5.24: Forwarder count for MGW with SSV channel

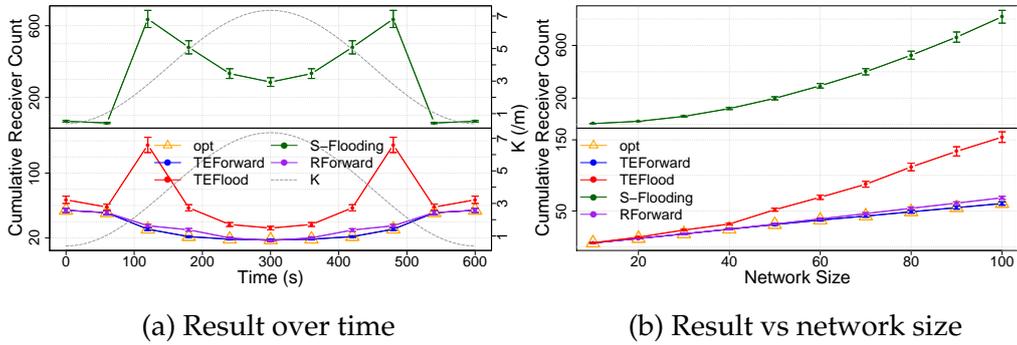


Figure 5.25: Cumulative receiver count for SGW with LSV channel

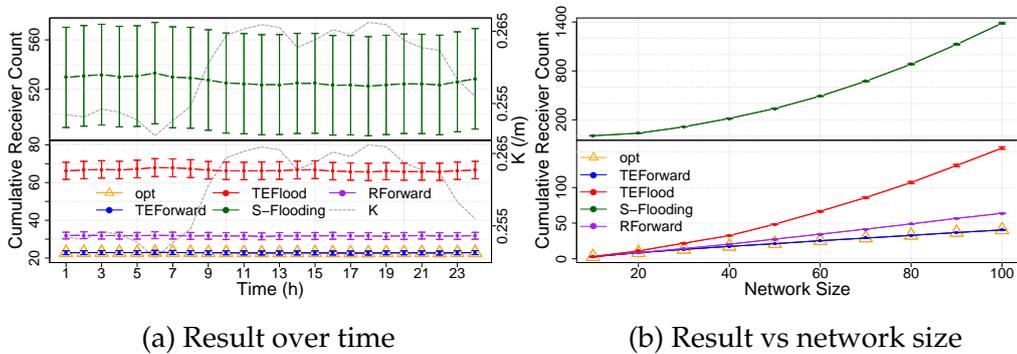


Figure 5.26: Cumulative receiver count for SGW with SSV channel

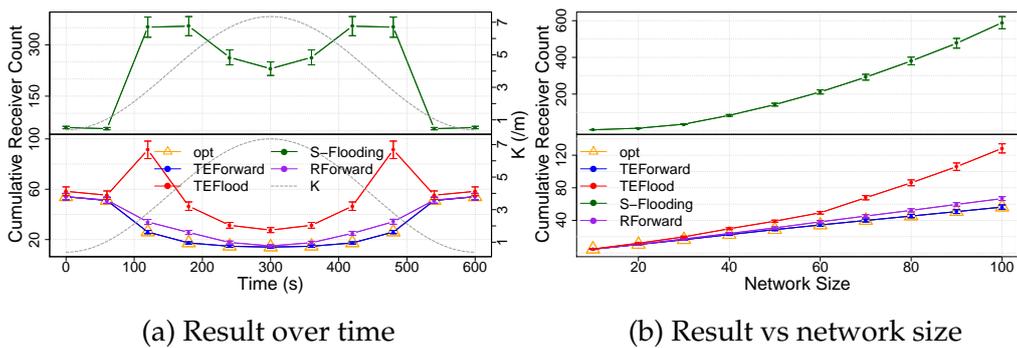


Figure 5.27: Cumulative receiver count for MGW with LSV channel

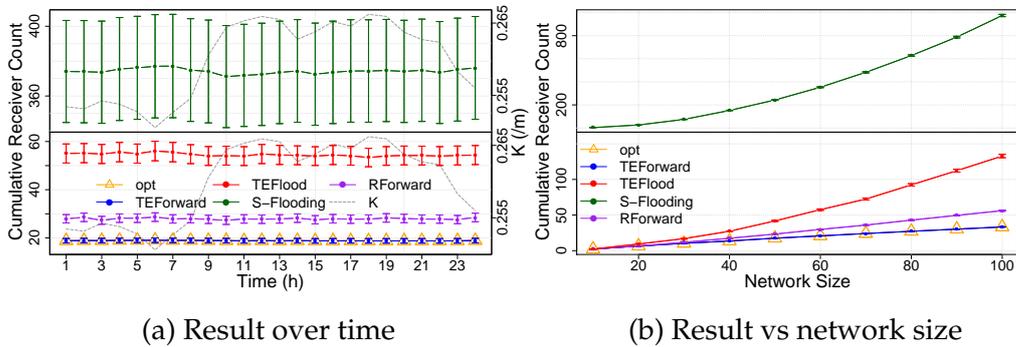


Figure 5.28: Cumulative receiver count for MGW with SSV channel

sults in more packet processing especially when the end-to-end hop count increases with a magnitude much higher than the decrease of node density, like the results at 120s and 480s. This phenomenon does not appear in SSV channel due to the low  $K$  that prevents high packet loss over time, as shown by the comparison between Fig. 5.5a and Fig. 5.6a.

### Fairness

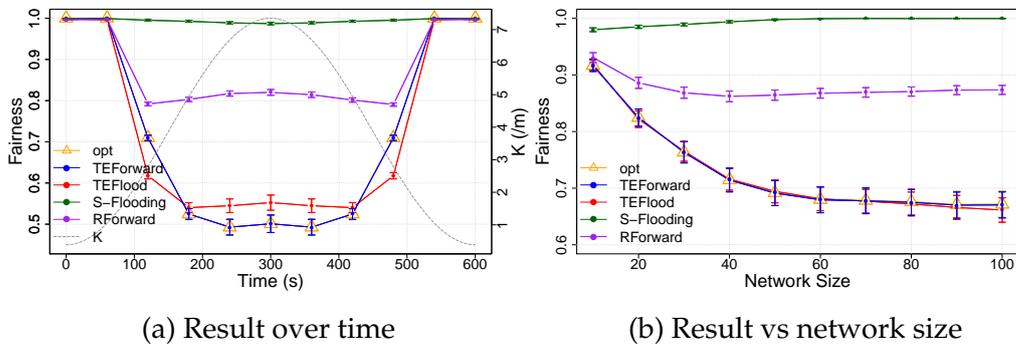


Figure 5.29: Fairness for SGW with LSV channel

TEForward and optimal forwarding achieve high energy efficiency at the expenses of fairness due to the path selection process that is greedy for energy efficiency. The same reason applies to TEFlood since the TTL-based

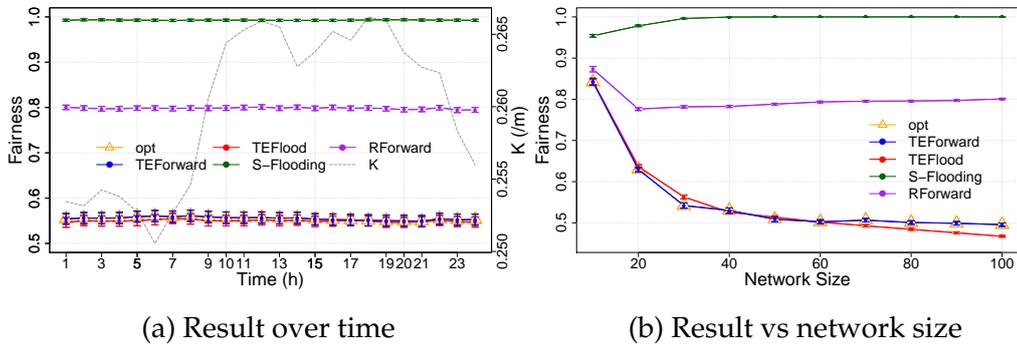


Figure 5.30: Fairness for SGW with SSV channel

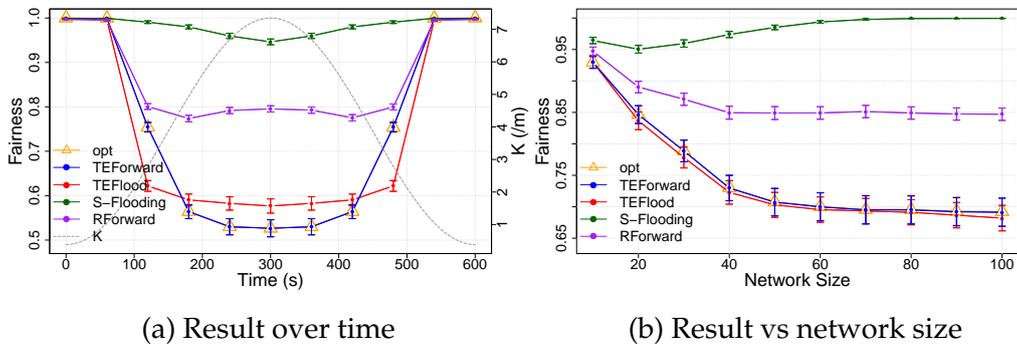


Figure 5.31: Fairness for MGW with LSV channel

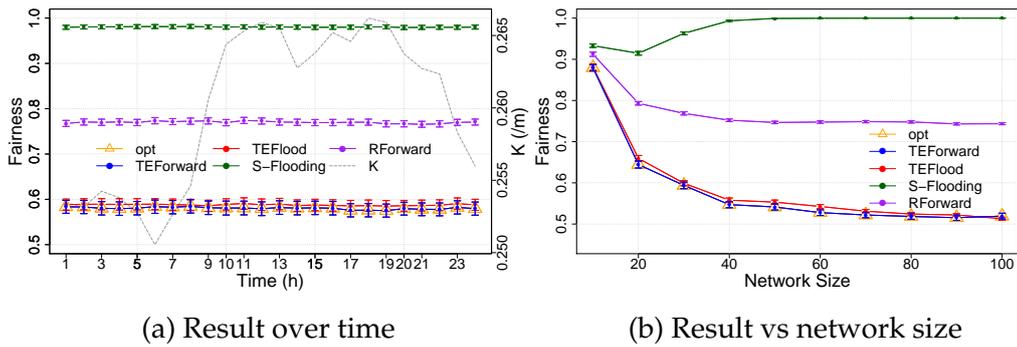


Figure 5.32: Fairness for MGW with SSV channel

packet direction unavoidably prioritizes certain nodes for forwarding. The S-Flooding and RForward show the highest and the second highest fairness because of the forwarding process with no preference for forwarders.

Summarizing the simulation results, by extracting the latest topology information from polling beacons, TEForward achieves high data delivery ratio and energy efficiency under dynamic channel conditions.

Table 5.3: Single-gateway LSV-channel performance summary

Metric	opt	TEForward	TEFlood	S- Flooding	RForward
$PDR$ (%)	99.8966	99.8908	99.8876	98.4127	81.0562
$P_{COLL}$ (%)	0.0583	0.0619	0.1105	0.1282	0.0690
$D$ (ns)	46.9352	64.8022	46.9386	46.6768	50.2398
$E$ (pJ)	2789.8314	2825.0502	5274.0766	57923.1505	10349.4733
$FD$	1	1	1.8248	10.7205	1.0015
$CR_{count}$	33.4646	33.5080	67.6898	311.8388	36.0401
$F$	0.7259	0.7253	0.7248	0.9943	0.8758

Table 5.4: Single-gateway SSV-channel performance summary

Metric	opt	TEForward	TEFlood	S- Flooding	RForward
$PDR$ (%)	99.8805	99.8738	99.9302	96.8593	76.0598
$P_{COLL}$ (%)	0.0187	0.0303	0.0877	0.1102	0.0391
$D$ (ns)	50.0113	68.9667	50.0156	49.5838	63.2951
$E$ (fJ)	23.0095	23.1389	59.1544	481.6851	50.4958
$FD$	1	1	2.2566	19.4703	1.0002
$CR_{count}$	22.7749	22.7783	66.4205	526.5526	31.8121
$F$	0.5552	0.5552	0.5500	0.9928	0.7983

Table 5.5: Multi-gateway LSV-channel performance summary

Metric	opt	TEForward	TEFlood	S- Flooding	RForward
$PDR$ (%)	99.9785	99.9772	99.9758	99.9917	87.7226
$P_{COLL}$ (%)	0.0520	0.0571	0.1007	0.1273	0.0652
$D$ (ns)	39.3770	55.2616	39.3786	39.3851	40.3564
$E$ (pJ)	2511.5705	2547.8571	4388.3077	42214.4636	7670.9805
$FD$	1	1	1.7593	9.1828	1.0829
$CR_{count}$	30.9036	30.9348	54.0989	220.3826	34.7491
$F$	0.7477	0.7472	0.7416	0.9822	0.8649

Table 5.6: Multi-gateway SSV-channel performance summary

Metric	opt	TEForward	TEFlood	S- Flooding	RForward
$PDR$ (%)	99.9658	99.9617	99.9803	99.9776	83.5316
$P_{COLL}$ (%)	0.0083	0.0161	0.0696	0.1165	0.0321
$D$ (ns)	41.7418	57.9904	41.7461	41.7506	47.3275
$E$ (fJ)	19.3151	19.4516	50.1479	343.7993	40.4077
$FD$	1	1	2.2972	16.0351	1.1564
$CR_{count}$	18.7689	18.8494	55.0957	373.7121	27.8205
$F$	0.5805	0.5805	0.5879	0.9803	0.7701

## 5.4 Summary

In this section, TEForward scheme is designed for the backhaul tier of multi-hop polling-based EM-WNSNs under dynamic channel states. The TEForward extracts the up-to-date topology information from polling beacons for forwarder selection and data diffusion with low overheads. Following the forwarding decisions of TEForward, all packets flow to the IoT gateway along the path with the minimum energy consumption determined at the moment of polling. From the performance evaluation, TEForward achieves higher data delivery ratio, high energy efficiency and low collision probability than the benchmarks.

## Chapter 6

# Bandwidth-aware Access for Event-driven IoNT

The OP polling discussed in Chapter 4 is not suitable for nanonetworks deployed for sensing bursty events under limited IoT bandwidth due to the receiver-initiated data collection. To achieve high resource utilization efficiency in this event-driven scenario, the Enhanced Adaptive Pulse Interval Scheduling (EAPIS) is designed. The EAPIS implements pulse-level transmission sequencing based on the allocated bandwidth so as to match the EM-WNSNs throughput with the IoT bandwidth. In this chapter, the ideal pulse arrival pattern is derived. Next, EAPIS is designed based on this ideal pulse pattern. Finally, the performance evaluation and modelling are conducted.

### 6.1 Enhanced Adaptive Pulse Interval

In this section, the ideal pulse arrival pattern at the EM-WNSN gateway and the design of EAPIS are discussed.

### 6.1.1 Bursty Events and Ideal Pulse Arrival

For EM-WNSNs connected to the Internet, matching the EM-WNSNs throughput to the available IoT bandwidth under bursty events is significant for the sake of resource utilization efficiency.

Here, bursty events for nano-sinks are defined as a group of events with event duration  $T_{event}^{total}$  shorter than the transmission duration of one sink.  $T_{event}^{total}$  is defined as

$$T_{event}^{total} = T_{event}^L - T_{event}^1, \quad (6.1)$$

where  $T_{event}^1$  and  $T_{event}^L$  are the time when the first event and the last event are reported to the nano-sinks, respectively.

The bursty events incur concurrent transmissions of multiple nano-sinks, which results in an aggregated throughput that exceeds the allocated bandwidth if the pulses are not ideally scheduled.

Given the allocated IoT bandwidth  $B$ , the pulse duration  $t_P$  and pulse interval  $I$ , to achieve the maximum data rate without triggering traffic regulation mechanisms,  $I$  is related with  $B$  using Eqn. (6.2):

$$\frac{1}{t_P + I} = B, \quad (6.2)$$

which allows us to calculate the shortest pulse interval  $I_S$ :

$$I_S = \frac{1}{B} - t_P. \quad (6.3)$$

The  $I_S$  leads to the ideal arrival pattern of pulses which maximally utilizes the IoT bandwidth with on-demand throughput, as shown in Fig. 6.1.

Pulses arriving at the gateway that are not separated by intervals of duration  $I_S$  contribute to poor energy efficiency of nanonetworks and poor utilization efficiency of the IoT bandwidth. On the one hand, for nano-devices, unnecessary energy is consumed when pulses arrive densely with intervals shorter than  $I_S$  only to be dropped at the gateway (by traffic

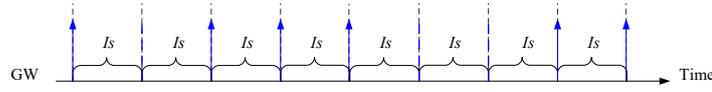


Figure 6.1: Ideal pulse arrivals at the gateway

policing mechanisms). On the other hand, for IoT access networks, sparse pulses with intervals longer than  $I_S$  introduce bandwidth efficiency. Based on the afore-mentioned observations, a scheme that adapts the pulse interval is devised to achieve both energy and bandwidth efficiency.

### 6.1.2 Overview of EAPIS

The EAPIS is a distributed algorithm run by nano-sinks in small-scale single-hop EM-WNSNs detecting bursty events. EAPIS algorithm adapts to the IoT bandwidth and neighbour degree and schedules the pulses transmitted from nano-sinks to the IoT gateway based on the ideal pulse arrival pattern. Since EAPIS operates on nano-sinks, the process that aggregates data from nanosensors to nano-sinks is highly abstracted in this thesis. The pseudocode of EAPIS is presented in Algorithm 7 with notations presented in Table 6.1.

The EAPIS adopts sequential multiple-user transmission based on the network information collected from short channel sensing. Specifically, when nano-sinks start transmitting pulses upon events happen, transmissions of nano-sinks are scheduled in sequence with each sink transmitting pulses following the ideal pulse pattern. Therefore, pulses arrive at the gateway in accordance with the ideal pattern.

The EAPIS operates on the following assumptions:

1. The network is one-hop connected.
2. Nano-sinks are preconfigured with the short interval  $I_S$ .
3. Nano-sinks are in receiving mode for channel sensing when bursty

events occur. This is a feasible operation for nano-sinks that are designed for data collection.

4. High time precision is required for scheduling. According to the existing research on nanonetworks [103, 119, 148, 149], this is feasible for nano-devices deployed in nano-scale scenarios with tiny propagation delay.

### 6.1.3 EAPIS Preamble



Figure 6.2: Packet structure

An EAPIS packet adopts a preamble containing “1” followed by “0” as shown in Fig. 6.2. The “1” in the beginning is only used for neighbour discovery and is transmitted directly in a burst. Eventually the preamble “1” is dropped. The following “0” is the real preamble and maintains the energy consumption of the EAPIS preamble as same as an original preamble. After receiving and dropping the preamble “1”, when it reaches the time to receive “0”, the IoT gateway inverses “0” to “1” and sends it to the access network as the first bit. In this way, neighbour discovery is accomplished without losing the first bit under traffic policing. With the introduction of this preamble, an EAPIS packet achieves the same energy consumption with marginal delay increase in comparison with the original packet.

### 6.1.4 Transmission Sequencing

To achieve the overall ideal pulse arrival pattern, in EAPIS, each nano-sink transmits its pulses following the ideal pattern and the transmissions of nano-sinks are scheduled to be sequential. The sequencing is implemented by making the preamble interval of each sink, which is the interval between the preamble pulse and the preamble "0", equal one  $I_S$  after the prior transmissions of neighbours.

Assuming the packet size is known and identical for each sink, the most important factor for obtaining the preamble interval for each sink becomes the number of neighbours transmitting prior to itself. This is achieved by counting the number of preamble pulses received from neighbours before the sink transmits its own preamble pulse, as denoted by  $S^k$ . For obtaining  $S^k$ , each sink senses the channel before it transmits the preamble pulse and stops sensing afterwards for saving energy.

When events happen, the first sink detecting the event transmits both the preamble and payload pulses with an interval of  $I_S$  because no neighbour is sensed, as presented by lines 1 - 5 and line 16 in the transmitting mode of Algorithm 7. When a nano-sink detects an event after the first sink, it firstly transmits the preamble pulse for updating the  $S$  values for its neighbours, records the transmission time of the preamble pulse as  $t_{TX}^{k1}$ , and deactivates the channel sensing following line 7 - 9 in the transmitting mode. Here, a binary variable  $TxCount^k$  is used to identify the transmission of the preamble pulse.

For any sink that transmits after the first sink, when it receives the preamble pulse of the first sink, the receiving time  $t_{RX}^{k1}$  is recorded and its  $S^k$  is incremented, as shown in lines 2 - 4 in the receiving mode of Algorithm 7. The time stamp  $t_{RX}^{k1}$  is used in two ways later: 1) the synchronization reference to align other sinks to the first sink; and 2) the unique token to recognize the pulses of the first sink. When a sink receives the  $j$ -th pulse after the first preamble pulse, it records the receiving time  $t_{RX}^{kj}$  and identifies if this pulse is a preamble pulse from other neighbours or a

Table 6.1: Notations for EAPIS of Nano-sink  $k$ 

Notation	Description
$t_{TX}^{ki}$	Time to transmit the $i$ -th pulse
$t_{RX}^{kj}$	Time when the $j$ -th neighbour pulse is received
$S^k$	The number of neighbours transmitting prior to a sink
$I^k$	The interval for the next pulse
$I_S$	The shortest pulse interval
$TxCOUNT^k$	The preamble pulse identifier
$P$	The packet size

payload pulse from the first sink that is transmitting. This is done by line 5 in the receiving mode which checks if the time interval between this pulse and the first preamble pulse is an integer multiple of  $I_S + t_P$  that equals the interval between two successive pulses received from the first sink. This operation is reliable for the femto-second-long pulses in EM-WNSNs benefiting from the low probability for the preamble pulses to collide with the first sink's payload pulses. Then, the nano-sink increments its  $S^k$  if the pulse is recognized as a preamble pulse.

After collecting the  $S^k$  statistics and transmitting the preamble pulse, the nano-sink, which transmits later than the first sink, calculates its preamble interval. The preamble interval given by line 10 in the transmitting mode is the sum of two parts: 1)  $I_S(S^k P - S^k + 1) + t_P S^k (P - 1)$  that is one  $I_S$  after  $S^k$  neighbour transmissions before the current sink, and 2)  $t_{RX}^1 - t_{TX}^1$  that aligns all other sinks to the first sink as mentioned above. After transmitting the preamble, the sink sets the pulse interval to  $I_S$  for rest of pulses in the packet to achieve the ideal pulse pattern.

---

**Algorithm 7** Enhanced Adaptive Pulse Interval Scheduling (EAPIS) for Nano-sink  $k$

---

**Initialization**

$$t_{TX}^{ki} = 0, t_{RX}^{kj} = 0, S^k = 0, I^k = I_S, TxCount^k = 0$$

**Transmitting mode**

- 1: **if** reaches the time  $t_{TX}^{ki}$  to transmit the  $i$ -th pulse **then**
- 2:   Transmit the pulse
- 3:   **if**  $S^k == 0$  **then**
- 4:      $I^k = I_S$
- 5:     Stop sensing channel
- 6:   **else**
- 7:     **if**  $TxCount^k == 0$  **then**
- 8:       Record  $t_{TX}^{k1}$
- 9:       Stop sensing channel
- 10:       $I^k = I_S(S^k P - S^k + 1) + t_P S^k (P - 1) + t_{RX}^{k1} - t_{TX}^{k1}$
- 11:       $TxCount^k = TxCount^k + 1$
- 12:     **else**
- 13:        $I^k = I_S$
- 14:     **end if**
- 15:   **end if**
- 16:   Transmit the next pulse after  $I^k$
- 17: **end if**

**Receiving mode**

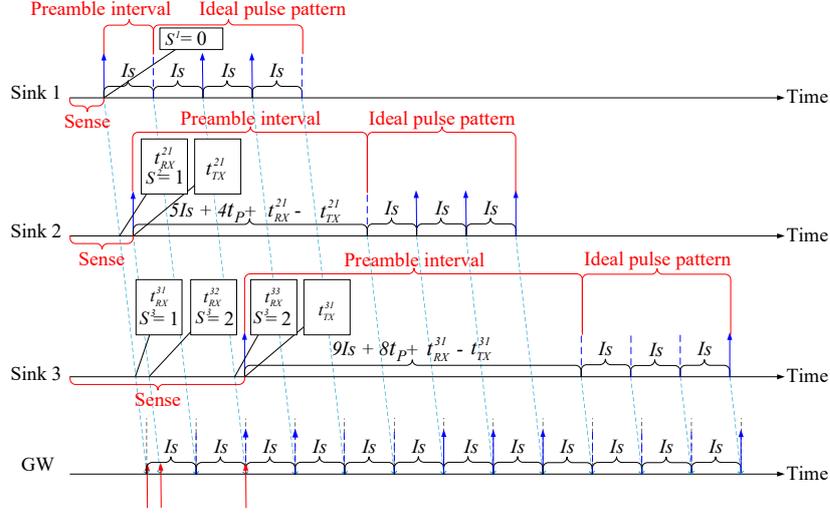
- 1: **if** Receive the  $j$ -th pulse at time  $t_{RX}^{kj}$  **then**
  - 2:   **if**  $j == 1$  **then**
  - 3:     Record  $t_{RX}^{k1}$
  - 4:      $S^k = S^k + 1$
  - 5:   **else if**  $(t_{RX}^{kj} - t_{RX}^{k1}) \bmod (I_S + t_P) \neq 0$  **then**
  - 6:      $S^k = S^k + 1$
  - 7:   **end if**
  - 8: **end if**
-

In this way, each nano-sink transmits pulses following the ideal pulse arrival and nano-sinks transmit one after another in a sequence.

The EAPIS implements pulse scheduling based on neighbour information. For the  $i$ -th sink transmitting on event occurrence, EAPIS shows linear time complexity  $\mathcal{O}(i - 1)$ , therefore it is suitable for resource-constraint nano-devices.

### 6.1.5 EAPIS Example

The full operations of EAPIS are depicted via the example in Fig. 6.3. Three nano-sinks that are Sink 1, Sink 2, and Sink 3 detect events and transmit pulses to the IoT gateway. Sink 1 transmits all its pulses following the ideal pulse pattern because  $S^1 = 0$  when it transmits the preamble pulse. When Sink 2 receives the preamble pulse at  $t_{RX}^{21}$  during channel sensing, it records  $t_{RX}^{21}$  and increments  $S^2$  indicating that Sink 1 is transmitting before itself. After transmitting its own preamble pulse at  $t_{TX}^{21}$ , Sink 2 stops sensing the channel and sets its preamble interval to  $5I_S + 4T_P + t_{RX}^{21} - t_{TX}^{21}$  based on line 10 in Algorithm 7 so that its future pulses with an interval of  $I_S$  will arrive at the gateway one  $I_S$  after the last pulse of Sink 1. Sink 3 operates in the same way except that  $S^3$  remains as 2 when it receives the third pulse, which is recognized as a payload pulse of Sink 1, during channel sensing. Overall, all pulses except the preamble pulses arrive at the gateway following the ideal pulse arrival.

Figure 6.3: Adaptive pulse interval Scheduling ( $N' = 3, N = 3$ )

## 6.2 Performance Evaluation

This section first defines the performance metrics used for performance evaluation and lists several benchmark schemes for comparison, followed by the simulation parameters. At last, performance results are presented and analysed.

### 6.2.1 Performance Metrics

1. Bandwidth efficiency,  $BE$ :

$$BE = \frac{\lambda_E}{B}, \quad (6.4)$$

where  $\lambda_E$  is the effective throughput that denotes the actual throughput after traffic policing. The  $BE$  is a key metric that measures the fitness of the effective throughput to the given bandwidth.

2. Pulse drop ratio,  $D$ :

$$D = \frac{N_{pulse}^D}{N_{pulse}}, \quad (6.5)$$

where  $N_{pulse}^D$  is the number of pulses dropped by traffic policing at the gateway and  $N_{pulse}$  is the total number of pulses received by the gateway. The  $D$  measures the unnecessary energy consumption due to discarded pulses.

3. Unit energy consumption,  $E$ :

$$E = \frac{E_{total}}{P_{total}}, \quad (6.6)$$

where  $E_{total}$  denotes the total energy consumption at nano-sinks transmitting pulses and  $P_{total}$  is the number of pulses effectively forwarded to the access network after traffic policing. The  $E$  measures the energy consumption for effectively receiving one data bit at the gateway.

4. First packet delay,  $T_{pkt}$ :

$$T_{pkt} = T_{pulse}^{pkt-1-L} - T_{event}^1, \quad (6.7)$$

where  $T_{pulse}^{pkt-1-L}$  is the time when the last pulse of the first packet is effectively received by the gateway. This metric measures the latency for the gateway to detect the event assuming that an event is identified in the packet level.

5. Pulse delay,  $T_{pulse}$ :

$$T_{pulse} = T_{pulse}^L - T_{event}^1, \quad (6.8)$$

where  $T_{pulse}^L$  is the receiving time for the last pulse of the total transmission and  $T_{event}^1$  is the detection time of the first event. This metric measures the total transmission time to receive all pulses.

6. Fairness,  $F$ :

$$F = \frac{(\sum_{i=1}^{N'} \lambda_E^i)^2}{N' \sum_{i=1}^{N'} \lambda_E^i}, \quad (6.9)$$

where  $\lambda_E^i$  is the effective throughput of the  $i$ -th sink. Fairness measures the distribution of effective throughput among sinks.

7. Packet delivery ratio,  $PDR$ :

$$PDR = \frac{N_{pkt}^E}{N_{pkt}}, \quad (6.10)$$

where  $N_{pkt}^E$  is the number of packets effectively forwarded to the access network and  $N_{pkt}$  is the total number of packets from nanonetworks. The access network fails to receive a packet if any pulse in the packet is dropped. The metric is significant for applications that demand packet-level processing.

### 6.2.2 Benchmarks

To benchmark EAPIS, several scheduling schemes are selected for comparison. They investigate the effects of neighbour awareness (with/without the knowledge of neighbour degree) and multiple-user arrival pattern (sequential/interleaved) on pulse arrival scheduling.

1. Adaptive Pulse Interval Scheduling (APIS): APIS implements interleaved transmission for multiple users to share the IoT bandwidth. It adopts the same packet preamble and assumptions as EAPIS. The basic operations of APIS are composed of transmission shifting and interleaving as shown in Algorithm 8.

The transmission shifting aims to sequence the transmissions of individual nano-sinks with an interval of  $I_S$  in alignment with the first sink transmitting. To reach this goal, for each sink except the first one, two local parameters are required. First, the transmission time of the global first pulse,  $t_1^k$ . Here, the global  $i$ -th pulse is defined as the  $i$ -th pulse among pulses from all sinks whereas the local  $i$ -th pulse denotes the  $i$ -th pulse from one particular sink. Second, the number of neighbours that access the channel prior to the current sink,  $S^k$ .

**Algorithm 8** Adaptive Pulse Interval Scheduling (APIS) for Nano-sink  $k$ **Transmitting mode**

- 1: **if** reaches the time  $t_{TX}^{ki}$  to transmit the  $i$ -th pulse **then**
- 2:   Transmit the pulse
- 3:   **if**  $TxCOUNT^k \leq 2$  **then**
- 4:      $TxCOUNT^k = TxCOUNT^k + 1$
- 5:   **end if**
- 6:   **if**  $TxCOUNT^k == 1$  **then**
- 7:     **if**  $S^k == 0$  **then**
- 8:        $I^k = I_L$
- 9:     **else**
- 10:        $I^k = t_1^k - t_{TX}^{k1} + I_L + S^k I_S$
- 11:     **end if**
- 12:   **else if**  $TxCOUNT^k == 2$  **then**
- 13:      $I^k = (ND^k + 1)I_S + ND^k t_P$
- 14:     Stop channel sensing
- 15:   **end if**
- 16:   Transmit the next pulse after  $I^k$
- 17: **end if**

**Receiving mode**

- 1: **if** Receive the  $j$ -th pulse at time  $t_{RX}^{kj}$  **then**
- 2:   **if**  $TxCOUNT^k == 0$  **then**
- 3:     **if**  $j == 1$  **then**
- 4:        $t_1^k = t_{RX}^{k1}$
- 5:     **end if**
- 6:      $S^k = S^k + 1$
- 7:   **end if**
- 8:    $ND^k = ND^k + 1$
- 9: **end if**

Transmission shifting begins with the transmission of the global first pulse. After transmitting this pulse, the first sink shifts its subsequent transmission by a long interval  $I_L$  that is expressed as:

$$I_L = NI_S, \quad (6.11)$$

where  $N$  is the total number of nano-sinks  $N$  that configured for sinks during network deployment.

$I_L$  ensures sufficient time for preamble exchange considering the potential time spread of events and the computational latency of nano-sinks, which is significant for successful scheduling. The gateway is also aware of  $I_L$  so as to detect the global first "0" and following "0"s in APIS preambles that are sequentially separated by  $I_S$ .

For each sink receiving the global first pulse at time  $t_{RX}^{k1}$ ,  $t_1^k$  is obtained by recording  $t_{RX}^{k1}$ , as shown by line 4 in the receiving mode of Algorithm 8. Inferring the transmission time of the global first pulse from the receiving process is reliable for small-scale EM-WNSNs with short propagation delay. Next, as in line 6 in the receiving mode,  $S^k$  for each sink  $k$  is obtained by counting the number of preamble "1"s received before transmitting the local first pulse, which is conditioned on  $TxCOUNT^k == 0$  whereby  $TxCOUNT^k$  is a two-bit long transmission counter.

After transmitting the preamble "1" at time  $t_{TX}^{k1}$ , a nano-sink will shift its subsequent pulses by an interval  $I^k$  given in line 10 in the transmitting mode of Algorithm 8. The interval  $I^k$  is the sum of three parts, namely,  $I_L$ ,  $t_{RX}^{k1} - t_{TX}^{k1}$ , and  $S^k I_S$ . The subsequent transmission of the current sink will first be aligned to the first sink via  $t_{RX}^{k1} - t_{TX}^{k1} + I_L$  and then shifted to be  $I_S$  later than the neighbour sink transmitting prior to it via  $S^k I_S$ .

After shifting the transmissions of all nano-sinks, subsequent pulses from all sinks following APIS preambles will be interleaved with an

interval that evenly shares the bandwidth. Assuming there are  $N'$  out of  $N$  sinks active for transmission, the bandwidth allocated for each sink becomes  $\frac{B}{N'}$ . By replacing  $B$  with  $\frac{B}{N'}$  in Eqn (6.2), the pulse interval  $I^k$  for subsequent pulses is derived in Eqn. (6.12):

$$I^k = \frac{N'}{B} - t_P = N' I_S + (N' - 1)t_P. \quad (6.12)$$

The value of  $N'$  is obtained by counting the number of “1”s in APIS preambles received during the channel sensing, which implies the neighbour degree  $ND^k$ , as presented by line 8 in the receiving mode of Algorithm 8. Finally, by replacing  $N'$  with  $ND^k + 1$  in Eqn. (6.12),  $I^k$  is updated in line 13 in the transmitting mode of Algorithm 8. As shown in the following steps, each nano-sink transmits subsequent pulses with the interval  $I^k$  and stops sensing the channel after the transmission time of the preamble “0” since sufficient information has been obtained.

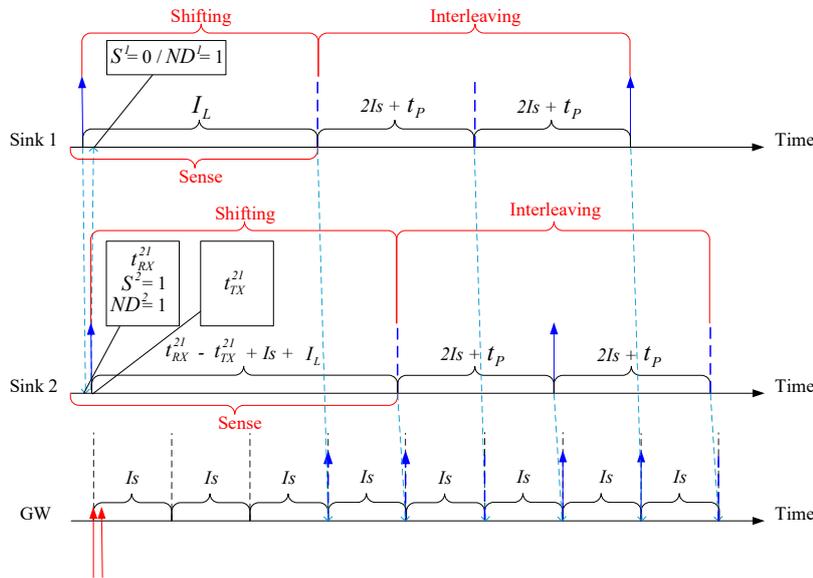


Figure 6.4: Adaptive pulse interval Scheduling ( $N' = 2, N = 3$ )

The full operations of APIS is depicted via the example in Fig. 6.4.

Two out of three nano-sinks, which are Sink 1 and Sink 2, are transmitting pulses to the IoT gateway. Sink 1 transmits the global first pulse and shifts its subsequent transmission by  $I_L$  time. When Sink 2 receives this pulse at  $t_{RX}^{21}$ , it approximates  $t_1^2$  with  $t_{RX}^{21}$  and increments both  $S^2$  and  $ND^2$  to 1 since Sink 1 transmits prior to it. After transmitting the local first pulse at  $t_{TX}^{21}$ , Sink 2 shift its next transmission by  $t_{RX}^{21} - t_{TX}^{21} + S^2 I_S + I_L$  time. By receiving the local first pulse of Sink 2, Sink 1 only increments  $ND^1$  to 1 since Sink 2 transmits after it. When it reaches the time for the local second pulses, which are "0"s in preambles, the pulse interval of both Sink 1 and Sink 2 are set to  $I^k = 2I_S + t_P$  which interleaves transmissions of two sinks with evenly shared bandwidth. Channel sensing is then stopped.

2. Long Pulse Interval Scheduling (LPIS): in LPIS, pulses are transmitted directly with a fixed interval  $I_L$ .
3. Random Pulse Interval Scheduling (RPIS): RPIS is similar to LPIS but with random pulse intervals. For each pulse to be transmitted, the interval is given by a uniform random number  $RI_S$  whereby  $R \sim U[1, N]$ .
4. Short Pulse Interval Scheduling (SPIS): SPIS originates from the basic unslotted Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA). Before transmitting the first pulse, the nano-sink conducts channel sensing for  $I_S$ . All pulses are then transmitted with an interval of  $I_S$  if the channel is clear. Otherwise, an one-packet-long backoff time is adopted before the next channel sensing.
5. Short Pulse Interval Scheduling with Random Backoff (SPIS-RB): SPIS-RB is SPIS with uniform random backoff time that is given by  $RI_S$  with  $R \sim U[0, P - 1]$  where  $P$  denotes the packet size.

Table 6.2: Simulation parameters

Parameter	Value
Frequency	100 GHz
Pulse energy $e_{TX}$	1 aJ
Pulse duration $t_P$	100 fs
Receiving sensitivity	-130 dBm
Packet size $P$	16 bits (17 bits for EAPIS/APIS)
Event time $t_{start}$	[0 $\mu$ s, 20 $\mu$ s]
Simulation area	10 cm $\times$ 10 cm $\times$ 10 cm
Network size $N$	10 - 50 with a step of 5
IoT gateway position	Geometric centre
Nano-sink position / mobility	Uniform / static
Molecular absorption coefficient $K$	2.6e-5 m <sup>-1</sup>
IoT bandwidth $B$	200 Kbps

### 6.2.3 Simulation Parameters

Performance of the pulse scheduling schemes is evaluated using the Nano-Sim package [145] on NS-3 with pulse-level simulations enabled. Simulation parameters are listed in Table 6.2. The adopted scenario comprises 10 to 50 nano-sinks randomly deployed following uniform distribution in a 10cm  $\times$  10cm  $\times$  10cm 3D space with the IoT gateway located at the geometric centre. To establish direct links among network devices, the following communication parameters are adopted. Each bit "1" is carried by a 100-fs-long pulse on 100 GHz with energy  $e_{TX}$  of 1 aJ and receiver sensitivity of -130 dBm based on the reported capacity of nano materials [3,4].

The molecular absorption coefficient  $K$  [15] that determines THz channel conditions is set to 2.6e-5m<sup>-1</sup> given by HITRAN (HIgh resolution TRANsmission) molecular absorption database ("USA Model, Mean Latitude, Summer, H=0") [9]. Each nano-sink has a 50% probability to transmit for event reporting when events occur considering that nano-devices are con-

figured for heterogeneous sensing tasks in reality [150]. To simulate nano-scale bursty events, in each simulation, a nano-sink starts transmitting one packet at a random starting time  $t_{start} \in [0\mu s, 20\mu s]$ . Each packet has a size  $P$  of 16 bits (17 bits for EAPIS/APIS) with a 50% probability of bit "1" [25]. The IoT bandwidth  $B$  is set to 200 Kbps following NB-IoT. Considering the low-cost feature of IoT devices, traffic policing activated every  $\frac{1}{B}$  time with a precision of nanoseconds is deployed at the gateway. Simulation finishes when all nano-sinks finish transmitting their pulses. The results are presented using mean values with 95% confidence intervals of 100 simulations on the basis of normally distributed errors. Each figure contains the full comparison and the magnified details of EAPIS/APIS at the bottom.

#### 6.2.4 Results and Analysis

In this section, the simulation results presented in Fig. 6.5 - 6.13 are discussed. Next, the mean values of results are summarized in Table 6.3.

##### Bandwidth Efficiency

To explore the pulse arrival pattern before traffic policing, one sample of raw throughput over time for a network with 20 sinks is shown in Fig. 6.5. As expected, EAPIS's throughput mostly matches the IoT bandwidth of 200 Kbps except for the spike in the beginning due to the bursty "1"s in EAPIS preambles. APIS exhibits the similar the pattern but underutilizes the bandwidth after the bursty preamble pulses due to the long waiting interval. In comparison, other benchmark schemes mostly mismatch the access bandwidth.

From the averaged bandwidth efficiency in Fig. 6.6, after traffic policing, EAPIS utilizes 99.40% of the bandwidth on average due to the fine-grained arrival pattern. Because of the first preamble pulse dropped, EAPIS only underutilizes the bandwidth during the first preamble interval that is

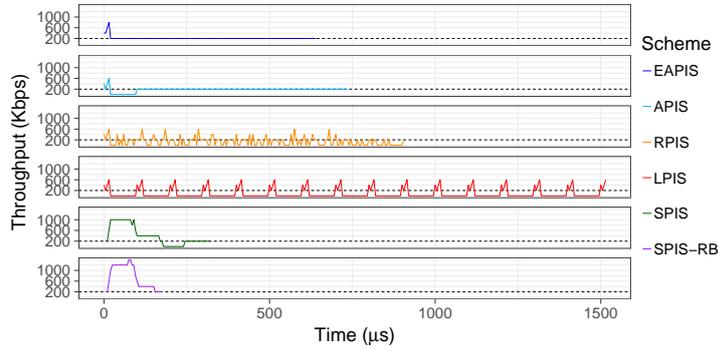
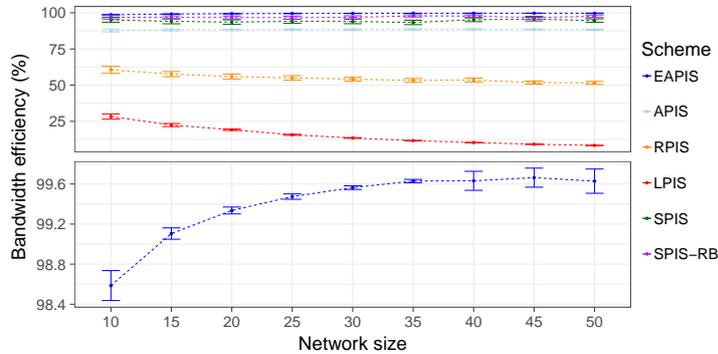
Figure 6.5: Raw throughput over time ( $N = 20$ )

Figure 6.6: Bandwidth efficiency vs network size

$I_S$ . Therefore, its bandwidth efficiency increases with the network size as a result of the reduced proportion of underutilized bandwidth in the prolonged total transmission. However, this trend changes slightly for large network sizes (e.g. network size  $\geq 40$  in Fig. 6.6) due to the pulse dropping discussed later. As comparison, APIS only achieves 88.32% due to the long waiting interval that is especially impactful for the short packet size adopted in the simulation.

Below EAPIS, SPIS and SPIS-RB achieve 94.38% and 97.06% efficiency, respectively, benefiting from the short pulse interval adopted and the medium access mechanism that actively competes for transmission opportunities. Unfortunately, this high efficiency comes from the raw throughput that ex-

ceeds the bandwidth, which happens due to the wrong estimation of channel states when neighbours are transmitting “0”s that cannot be sensed. LPIS shows the lowest bandwidth efficiency due to two reasons. First, the interval  $I_L$  shares  $B$  among the total number of nano-sinks rather than the ones transmitting, which underutilizes  $B$ . Second, simply confining transmission speed does not suffice to implement the ideal arrival pattern.

**Pulse Drop Ratio**

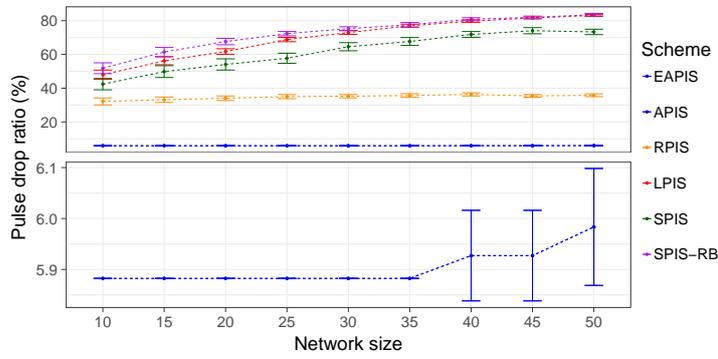


Figure 6.7: Pulse drop ratio vs network size

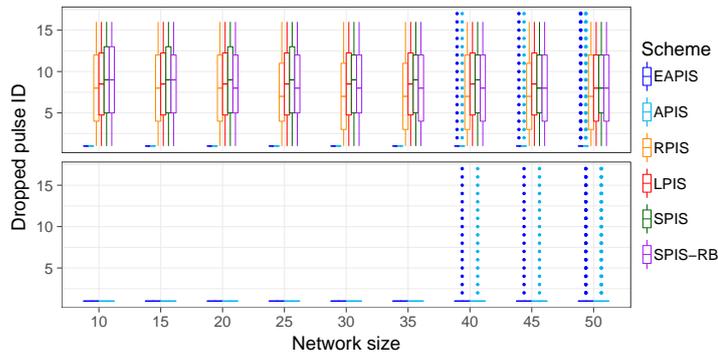


Figure 6.8: Dropped pulse ID vs network size

The pulse drop ratio is presented in Fig. 6.7. Under traffic policing, EAPIS and APIS mostly only lose the first pulses in preambles, which is

shown in Fig. 6.8 that depicts the ID distribution of the pulses dropped. Therefore, the pulse drop ratio is mostly consistent as 5.88% given by  $\frac{N'}{N'P_{size}}$ . However, the throughput of EAPIS and APIS may exceed the given bandwidth and trigger traffic policing for large network sizes (e.g. network size  $\geq 40$  in Fig. 6.7 and Fig. 6.8) due to the phenomenon that two sinks' prorogation delay is longer than the time separation of their events. In this circumstance, a nano-sink transmits its preamble pulse before receiving the preamble pulse of the neighbour transmitting prior to itself. Consequently, it fails to detect this neighbour, wrongly estimates its preamble interval, and overlaps its pulses with the neighbour in time domain. Fortunately, the probability and impact of neighbour detection failure is low for EAPIS and APIS considering the small coverage with short propagation delay of EM-WNSNs, which is demonstrated by the ID distribution that is highly concentrated at "1" with others identified as outliers for network size of 40, 45, 50 in Fig. 6.8.

Other benchmarks all demonstrate pulse drop ratio much higher than EAPIS and APIS. The SPIS-based schemes exhibits high pulse drop ratio as a result of the fixed interval and the false channel estimation that jointly result in consecutive pulse dropping. The RPIS achieves the second lowest pulse drop ratio since the random arrival pattern reduces the probability of two pulses consistently arriving within  $I_S$ . Again, by only changing data rate without pulse scheduling, LPIS experiences high pulse drop ratio.

### Unit Energy Consumption

As shown in Fig. 6.9, benefiting from the high bandwidth efficiency and low pulse drop ratio, EAPIS exhibits the lowest unit energy consumption close to the theoretically lowest level that is 0.5 aJ given that each packet contains 50% bit "1". EAPIS consumes less energy than APIS because sinks only sense the channel before transmitting the preamble pulses whereas channel sensing is required by APIS through the long waiting in-

terval. However, for both EAPIS and APIS, the preamble indeed gives rise to more unit energy consumption due to the number of preambles “1”s received that increases quadratically with the network size, as modelled by Eqn. (6.24) in the next section. The high energy efficiency extends the network lifetime and relaxes the demand for energy providers especially when energy harvesting is applied on nano-devices.

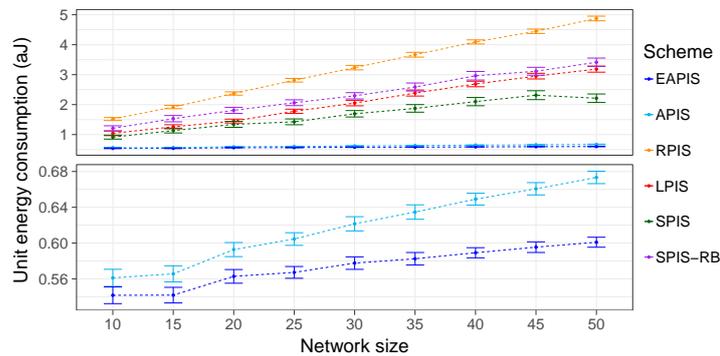


Figure 6.9: Unit energy consumption vs network size

### First Packet Delay

From Fig. 6.10, EAPIS that adopts sequential transmission with the long waiting interval eliminated achieves the lowest delay to effectively transmit the first packet. This feature is significant for the sensing applications which take packets as message data unit to achieve real-time responses. SPIS and SPIS-RB present similar performance as EAPIS because those schemes prioritize the first sink transmitting to be contention-free whereas let the following sinks compete for the channel. In comparison with EAPIS, the first packet delay for APIS is even higher than that of RPIS due to its long waiting interval that is linearly correlated to the network size.

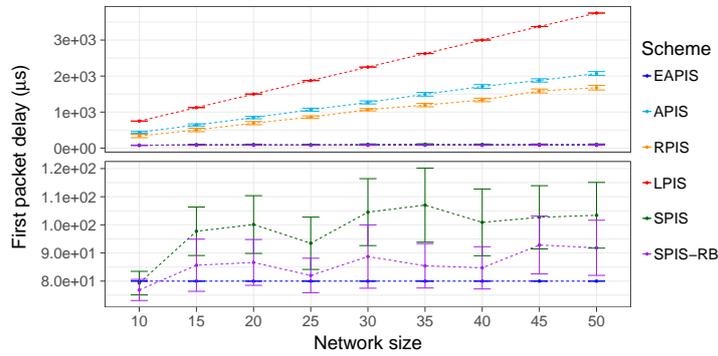


Figure 6.10: First packet delay vs network size

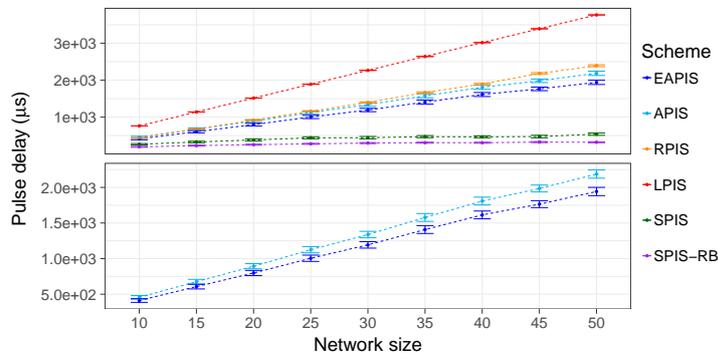


Figure 6.11: Pulse delay vs network size

### Pulse Delay

The total pulse delay is shown in Fig. 6.11. As expected, LPIS shows the highest delay due to its low data rate. Although SPIS and SPIS-RB present low pulse delay, the over-utilized bandwidth leads to high pulse drop ratio. Because of the eliminated long waiting interval, EAPIS further improves the latency based on APIS.

### Fairness

Fairness shown in Fig. 6.12 is positively correlated with pulse drop ratio discussed above. This is because that when pulses from different senders

arrive within  $I_S$ , only the pulse of the first sender is retained with others dropped. EAPIS and APIS both achieve high fairness because pulses from all sinks are effectively received by the gateway unless the aforementioned neighbour detection failure happens and leads to pulse dropping for some sinks.

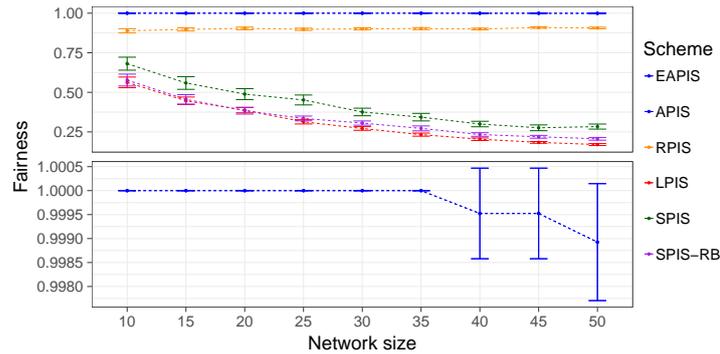


Figure 6.12: Fairness vs network size

### Packet Delivery Ratio

Discarding pulses leads to packet loss in the absence of retransmission and advanced error correction mechanisms. Fig. 6.13 shows that in most cases without neighbour detection failure, EAPIS and APIS achieve 100 % packet delivery ratio as a result of no pulse loss except the “1”s in preambles, which implies retransmission-free data delivery. This feature is significantly beneficial for nano-devices with constrained energy capacity. Compared with EAPIS and APIS, packet loss is observed for all benchmark schemes. It is worth noticing that there is no direct relationship between pulse drop ratio and packet loss ratio since the latter is determined by the pulse loss distribution among packets.

Overall, in comparison with those benchmarks, EAPIS achieves the best performance for all the performance metrics.

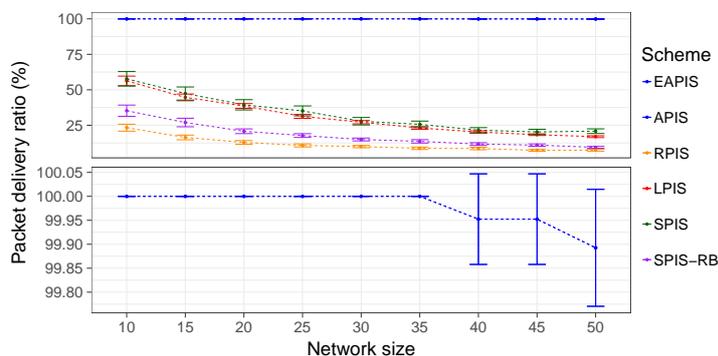


Figure 6.13: Packet delivery ratio vs network size

Table 6.3: Performance summary

Metric	EAPIS	APIS	RPIS	LPIS	SPIS	SPIS-RB
$BE$ (%)	99.4020	88.3204	54.8381	15.2360	94.3778	97.0640
$D$ (%)	5.9035	5.9035	34.7624	69.9154	61.7261	72.4984
$E$ (aJ)	0.5732	0.6181	3.2160	2.0884	1.6685	2.3320
$T_{pkt}$ ( $\mu$ s)	81.2124	1270.4775	1031.4003	2251.1845	85.8080	93.3018
$T_{pulse}$ ( $\mu$ s)	1193.5115	1338.5115	1413.9633	2266.7088	423.1474	279.2671
$F$	0.9998	0.9998	0.9008	0.3080	0.4176	0.3321
$PDR$ (%)	99.9775	99.9775	11.3740	30.2927	31.0249	16.0887

## 6.3 Performance Modelling

In this section, the performance models of bandwidth efficiency and unit energy consumption are developed and validated to serve as a guideline for network configurations when EAPIS and APIS are applied.

### 6.3.1 EAPIS Bandwidth Efficiency

To obtain the bandwidth efficiency  $BE$  in Eqn. (6.4), it is required to obtain the effective throughput  $\lambda_E$  given by

$$\lambda_E = \frac{1}{\Delta I_E}, \quad (6.13)$$

where  $\Delta I_E$  is the average pulse interval of the pulses effectively transmitted to the IoT infrastructure.

For EAPIS, because the first pulses in preambles are not retained,  $\Delta I_E$  can be obtained as

$$\Delta I_E = \tau / (P_{total} - 1), \quad (6.14)$$

where  $\tau$  is the total transmission time since the first preamble pulse is transmitted and  $P_{total}$  denotes the total number of pulses that are effectively received by the gateway.

The total sequential transmission is composed of the globally first preamble pulse and all packets with their preamble pulses omitted. Therefore,  $\tau$  is calculated as

$$\tau = t_P + I_S + I_S(P_{total} - 1) + t_p P_{total}. \quad (6.15)$$

The total number of effective pulses is obtained by subtracting the preamble pulses from packets, as expressed as

$$P_{total} = (P - 1)N', \quad (6.16)$$

where  $N'$  is the number of sinks transmitting given by

$$N' = \alpha N, \quad (6.17)$$

where  $\alpha$  denotes the probability that one sink involves in pulse transmission. Here,  $\alpha$  is set to a constant of 0.5. By substituting Eqn. (6.13) - (6.17) into Eqn. (6.4),  $BE$  for EAPIS is obtained as

$$BE = \frac{(P - 1)\alpha N - 1}{(P - 1 + Bt_P)\alpha N}. \quad (6.18)$$

From the final expression in Eqn. (6.18), the network size and the packet size are the two influencing factors for the bandwidth efficiency of EAPIS. Larger network sizes and packet sizes lead to higher bandwidth efficiency due to the reduced proportion of bandwidth underutilized during the preamble transmission of the first sink. Fig. 6.14 presents the

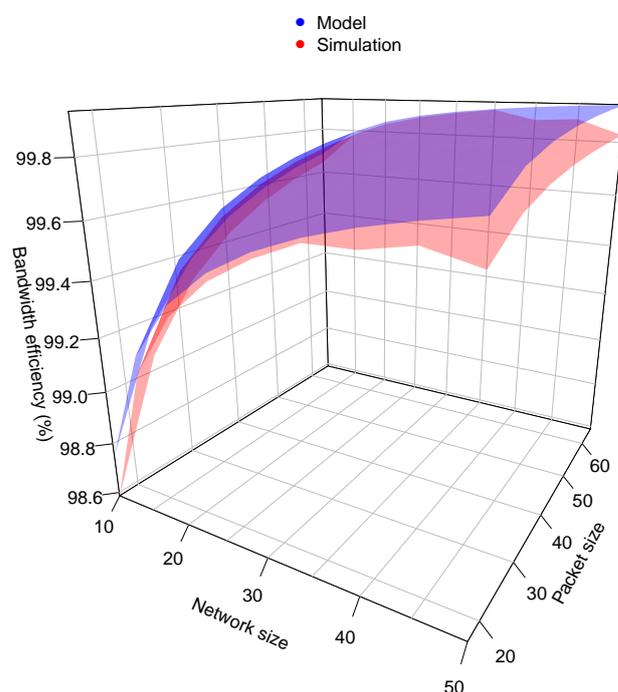


Figure 6.14: Bandwidth efficiency modelling for EAPIS (Model: blue surface; Simulation: red surface)

comparison between the model and simulation. It is worth noting that the modelling error increases as the network size increases because of the neighbour detection failure that results in interleaved transmission which leads to pulse dropping, as discussed in Section 6.2.4. acro

### 6.3.2 APIS Bandwidth Efficiency

By summing all pulse durations, the long waiting interval, and the total short pulse intervals in transmission,  $\tau$  for APIS is calculated as

$$\tau = t_P P^{total} + I_L + I_S(P^{total} - 1), \quad (6.19)$$

By substituting Eqn. (6.3), Eqn. (6.11), and Eqn. (6.13) - (6.17) into Eqn.

(6.4),  $BE$  for APIS is expressed as

$$BE = \frac{(P - 1)\alpha N - 1}{Bt_P(1 - N) + (P - 1)\alpha N - 1 + N}. \quad (6.20)$$

It can be seen that  $BE$  is proportional to  $P$ , which is validated through simulations with various  $P$  from 17 to 65 with a step of 8, as shown in Fig. 6.15. This is because that a long packet size mitigates the effect of idle channel during the APIS prefix.

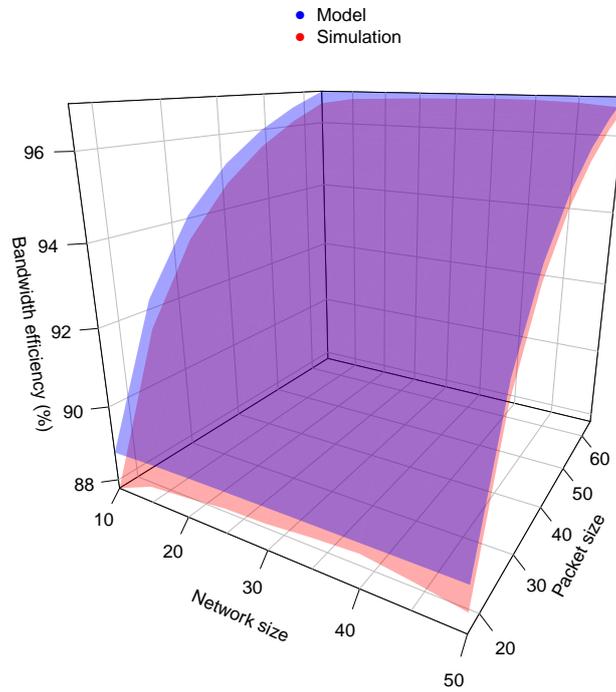


Figure 6.15: Bandwidth efficiency modelling for APIS (Model: blue surface; Simulation: red surface)

### 6.3.3 EAPIS Unit Energy Consumption

The total energy consumption of the nano-sinks transmitting pulses  $E_{total}$ , which is required for obtaining  $E$  in Eqn. (6.6), is given as

$$E_{total} = E_{total}^{TX} + E_{total}^{RX}, \quad (6.21)$$

where  $E_{total}^{TX}$  and  $E_{total}^{RX}$  are the total energy consumption for transmitting and receiving, respectively.

Considering that one EAPIS packet is composed of a 2-bit preamble that is "10" and the remaining  $P - 2$  bits,  $E_{total}^{TX}$  can be expressed as

$$E_{total}^{TX} = e_{TX} N' (\beta(P - 2) + 1), \quad (6.22)$$

where  $\beta$  is the ratio of bit "1" to bit "0" that is set to 50% in this thesis.

For EAPIS, the first sink transmits immediately when an event is detected and a sink only listens to the channel before transmitting its preamble pulse. Therefore, the pulses received by a sink may contain: 1) the preamble pulses from the sinks that start transmission prior to this sink, and 2) the first sink's payload pulses which arrive before the sink transmits its first preamble pulse. Thus,  $E_{total}^{RX}$  is expressed as

$$E_{total}^{RX} = E_{preamble}^{RX} + E_{payload}^{RX}, \quad (6.23)$$

where  $E_{preamble}^{RX}$  and  $E_{payload}^{RX}$  are the energy consumed for receiving preamble pulses and payload pulses which are received before sinks transmit the preamble pulses, respectively.

For  $E_{preamble}^{RX}$ , the total number of preamble pulses received is expressed as the arithmetic series from 0 to  $N' - 1$ . Consequently,  $E_{preamble}^{RX}$  is given by

$$E_{preamble}^{RX} = e_{RX} \frac{N'(N' - 1)}{2}, \quad (6.24)$$

where  $e_{RX}$  is the energy consumption for receiving one pulse.

For each payload pulse of the first sink, the number of receiving operations triggered equals the number of preamble pulses transmitted after this payload pulse. This is obtained by subtracting the number of preamble pulses transmitted before this payload pulse from the total number of preamble pulses except the first one that belongs to the first sink itself. If  $n_p$  denotes the average number of bits (regardless of preamble or payload) of the first sink transmitted during the event duration, and  $n_a$  is the average number of preamble pulses transmitted during one interval between

two payload pulses,  $E_{payload}^{RX}$  can be expressed as

$$E_{payload}^{RX} = \begin{cases} e_{RX} \sum_{i=3}^{n_p} N' - 1 - (i-1)n_a, & \text{if } n_p \geq 3 \\ 0, & \text{otherwise} \end{cases} \quad (6.25)$$

where  $i$  starts from 3 because the first two bits of the first sink are preambles. When  $n_p < 3$ ,  $E_{payload}^{RX}$  equals 0 because all sinks have transmitted their preamble pulses and stopped channel sensing before the first sink transmits its payload.

In this thesis, the event time for each sink is assumed to follow uniform distribution with a minimum of 0 and a maximum of  $T_{event}^{total}$ , which leads to  $n_p$  given by

$$n_p = \left\lfloor \frac{T_{event}^{total} - T_{event}^1}{\frac{1}{B}} \right\rfloor + 1, \quad (6.26)$$

where  $T_{event}^1$  is the average event time for the first sink that can be expressed as

$$T_{event}^1 = \frac{T_{event}^{total}}{N' + 1}. \quad (6.27)$$

For the uniformly distributed event time,  $T_{event}^1$  also equals the average time distance between neighbour events. Thus,  $n_a$  can be obtained as

$$n_a = \frac{\frac{1}{B}}{T_{event}^1}. \quad (6.28)$$

Using the common assumption that the energy consumed for receiving one pulse is 10% of the amount consumed for transmitting one pulse [16],  $e_{RX}$  is related with  $e_{TX}$  as below:

$$e_{RX} = \gamma e_{TX}, \quad (6.29)$$

where  $\gamma = 10\%$ .

Finally, by substituting Eqn. (6.16), Eqn. (6.17), and Eqn. (6.21) - (6.29) into Eqn. (6.6),  $E$  for EAPIS is expressed in Eqn. (6.30). The modelled unit energy consumption is compared with the simulation in Fig. 6.16. On the one hand,  $E$  rises when network size increases because of the increased

energy consumption for receiving preamble pulses. On the other hand,  $E$  decreases as a result of the increased packet size due to the increased number of pulses that share the energy consumed by receiving preamble pulses.

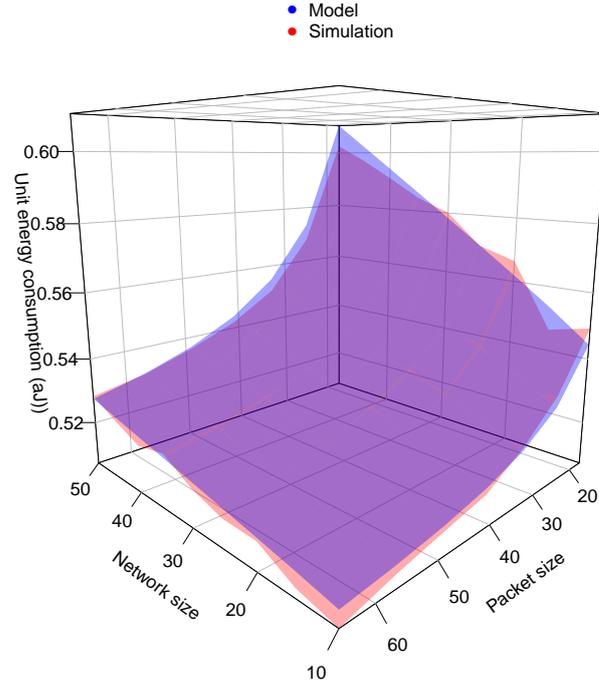


Figure 6.16: Unit energy consumption modelling for EAPIS (Model: blue surface; Simulation: red surface)

$$E = \begin{cases} \frac{e_{TX}}{\alpha N(P-1)} \left( \alpha N(\beta(P-2) + 1) + \gamma \left( \frac{\alpha N(\alpha N - 1)}{2} + \left( \lfloor \frac{BT_{event}^{total} \alpha N}{\alpha N + 1} \rfloor - 1 \right) \left( \alpha N - 1 - \frac{(\lfloor \frac{BT_{event}^{total} \alpha N}{\alpha N + 1} \rfloor + 2) (\frac{\alpha N + 1}{BT_{event}^{total}})}{2} \right) \right) \right), & \text{if } n_p \geq 3 \\ \frac{e_{TX}(\beta(P-2) + 1 + \frac{\gamma(\alpha N - 1)}{2})}{P-1}. & \text{otherwise} \end{cases} \quad (6.30)$$

### 6.3.4 APIS Unit Energy Consumption

For APIS, pulse receiving only happens during the sensing period before the transmission time of the second bit in one APIS preamble. Therefore,  $E_{total}^{RX}$  is given by

$$E_{total}^{RX} = e_{RX} N' (N' - 1), \quad (6.31)$$

where  $e_{RX}$  is the energy consumption for receiving one pulse.

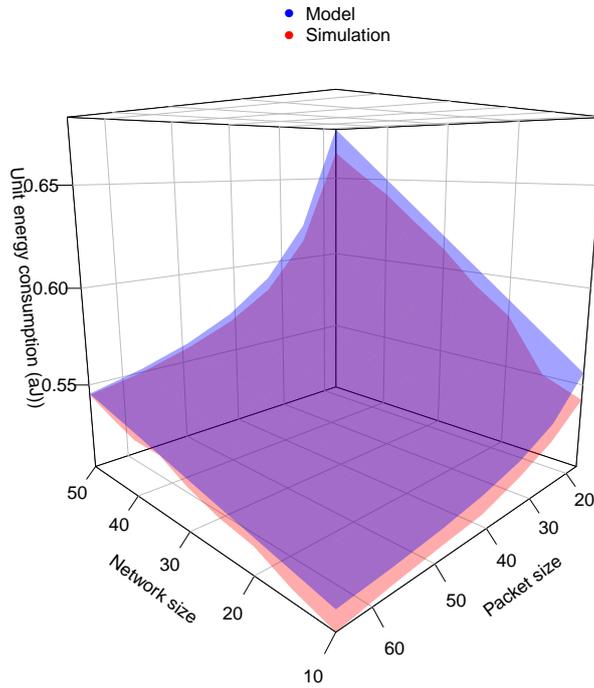


Figure 6.17: Unit energy consumption modelling for APIS (Model: blue surface; Simulation: red surface)

Eventually, by substituting Eqn. (6.16), Eqn. (6.17), and Eqn. (6.21) - (6.29) into Eqn. (6.6),  $E$  for APIS can be expressed as

$$E = e_{TX} \left( \frac{\beta(P-2) + \gamma(\alpha N - 1) + 1}{P-1} \right). \quad (6.32)$$

Based on the final model, packet size and total number of sinks are the influencing parameters for unit energy consumption. Therefore, the comparison between the analytical model and simulation results for various  $P$

and  $N$  are shown in Fig. 6.17. As discussed in section 6.2.4, unit energy consumption is impacted by the energy consumed by the APIS preamble. Therefore, increasing the  $P$  reduces the percentile of preamble in the total transmission, which decreases  $E$ . On the contrary,  $E$  increases with  $N$  since the number of preambles received increases quadratically with the network size.

## 6.4 Summary

To achieve high resource utilization efficiency, a distributed lightweight pulse scheduling scheme, namely EAPIS, is designed for event-based nanonetwork. Adopting simple operations with low complexity, EAPIS schedules pulses following the ideal arrival pattern when events happen. Overall, EAPIS achieves high energy efficiency for nanonetworks and high utilization efficiency for the IoT bandwidth.

# Chapter 7

## Conclusions and Future Work

The fast development of nano technologies is envisaged to enable nano-devices with integrated sizes in the level of micrometers. Networked nano-devices that communicate via THz electromagnetic waves form EM-WNSNs as a significant part of IoNT. Because of the non-invasiveness and high sensitivity of nano-scale devices, IoNT enables evolutionary applications that cannot be implemented via traditional WSNs. Nevertheless, data acquisition and dissemination for IoNT face challenges that originate from not only EM-WNSNs but also the existing Internet infrastructure. There are three major challenges that need to be addressed: the low capacity of nano-devices, the dynamic channel conditions of the THz band, and the limited and dynamic bandwidth of the IoT access and backhaul networks.

Therefore, this thesis contributes to layer 2 and layer 3 networking solutions that increase the resource utilization efficiency of IoNT taking into consideration the major challenges in the network paradigm.

### 7.1 Review

In Chapter 2, the overview of IoNT, existing work of MroAC layer, network layer in EM-WNSNs, and the polling solutions in WSNs are reviewed. The literatures show that limited research has been done for layer

2 and layer 3 IoNT under the constraints of the dynamic channel and IoT bandwidth. Next, in Chapter 3, the preliminary study on CForward for THz networks is discussed to improve the utilization efficiency of the THz band. Considering the complexity of CForward, lightweight networking mechanisms, which include OP polling, TEForward, and EAPIS, are discussed in Chapter 4, 5, and 6. Those solutions achieve high resource utilization efficiency for data acquisition and dissemination for IoNT.

## 7.2 Contributions

The contributions of this thesis are summarized below:

- CForward for THz networks is designed and evaluated. CForward is the first forwarding solution that considers the distance-dependent bandwidth of the THz band.
- OP polling that is the first solution for receiver-initiated data collection under dynamic THz channel conditions and IoT bandwidth in IoNT.
- TEForward which is an energy efficient end-to-end forwarding for polling-based IoNT under dynamic THz channel environments .
- EAPIS is designed, evaluated, and modelled. It is the first solution that enables efficient sender-initiated data collection for IoNT under limited IoT bandwidth .

## 7.3 Future Work

- **Polling with low latency and high robustness:** The current OP polling extracts necessary information for calculating the optimal packet aggregation size based on four handshakes, twice as many as the basic

polling, which results in almost doubled polling delay. This shortcoming could be mitigated by introducing predictive functions, which predict the network size and IoT bandwidth based on empirical data, to the IoT gateway. Potential time-series predictive methods such as autoregressive (AR), moving average (MV), and autoregressive integrated moving average (ARIMA) could be tested.

In addition, mechanisms that increase robustness in application scenarios with heterogeneous channel conditions shall be introduced for the sake of robustness. For example, IoNT is deployed for home monitoring whereby different rooms exhibit different humidity levels. In this case, nodes deployed in rooms such as the bathroom and kitchen with potentially high humidity should respond to polling with more trials considering the low link quality.

- **Polling for multi-gateway networks** Deploying multiple IoT gateways for data collection shows two benefits. First, for some applications, a full view of the sensed data is needed to achieve deep insights, which requires as many data as possible to be polled within the delay constraint. Multiple gateways could provide the required bandwidth assuming the path diversity of the access and backhaul network. Second, deploying multiple gateways could partition EMWNSNs. As a result, network performance could be improved due to the reduced end-to-end distance.

Nevertheless, polling for multi-gateway IoNT faces several challenges that hinder the application of the original OP polling. First, duplicate packets might be polled by multiple gateways, which unnecessarily consumes bandwidth and reduces data integrity. Second, network partition is determined not only by the distance from sinks to gateways but also by the bandwidth of each IoT. There would be trade-offs between bandwidth efficiency and energy efficiency (e.g. a gateway with high bandwidth covers more sinks to achieve high band-

width efficiency, however, sinks far away from the gateway would lead to high end-to-end cost for data delivery). Therefore, nano-sinks should be able to dynamically and pro-actively select the destination sink.

- **Pulse arrival scheduling for multi-gateway networks:** Considering the significance of deploying multiple gateways in IoNT, pulse arrival scheduling for sender-initiated data collection must be modified to support this new network architecture. For this purpose, the potential challenges are clarified as follows. First, since data are transmitted from sinks to gateways directly, multiple gateways may receive the same data. Therefore, mechanisms are to be designed for gateways to coordinate with one another for duplicate removal. Second, since EAPIS relies on preambles to determine transmission order of neighbours, grouping sinks before transmitting preambles is necessary.
- **Hybrid data acquisition for IoNT:** To support heterogeneous traffic pattern demanded by different user applications, multiple data collection modes could coexist in one network. For example, the same network could be deployed to serve different user applications such as periodic monitoring and event detection, which requires dynamic switching between receiver-initiated data collection and sender-initiated data collection. The coexistence of subscribed services imposes new challenges emerging from the conflict between services. One of the most important performance metrics that must be carefully considered is the energy efficiency of nano-devices. Consistently responding to adjacent services in time might rapidly drain the batteries of nodes. Therefore, buffering the sensed data at the gateways and sharing them with other delay-tolerant applications might be a solution for reducing energy consumption. Besides, different priorities could be allocated to services so as to ensure the ser-

vice quality for applications with high demands. Finally, resource allocation should be considered to mitigate the conflict between services.

# Bibliography

- [1] H. Yu, B. Ng, and W. K. G. Seah, "On-demand probabilistic polling for nanonetworks under dynamic iot backhaul network conditions," *IEEE Internet of Things Journal*, vol. 4, pp. 2217–2227, Dec 2017.
- [2] H. Yu, B. Ng, and W. K. G. Seah, "Pulse arrival scheduling for nanonetworks under limited iot access bandwidth," in *2017 IEEE 42nd Conference on Local Computer Networks (LCN)*, pp. 18–26, Oct 2017.
- [3] H. Yu, B. Ng, W. K. G. Seah, and Y. Qu, "Ttl-based efficient forwarding for the backhaul tier in nanonetworks," in *2017 14th IEEE Annual Consumer Communications Networking Conference (CCNC)*, pp. 554–559, Jan 2017.
- [4] H. Yu, B. Ng, and W. K. G. Seah, "On-demand efficient polling for nanonetworks under dynamic IoT backhaul network conditions," in *2016 IEEE 35th Int. Perform. Comput. Commun. Conf.*, pp. 1–8, dec 2016.
- [5] H. Yu, B. Ng, and W. K. G. Seah, "Forwarding schemes for em-based wireless nanosensor networks in the terahertz band," in *Proceedings of the Second Annual International Conference on Nanoscale Computing and Communication, NANOCOM' 15*, (New York, NY, USA), pp. 17:1–17:6, ACM, 2015.

- [6] I. F. Akyildiz and J. M. Jornet, "Electromagnetic wireless nanosensor networks," *Nano Commun. Netw.*, vol. 1, no. 1, pp. 3–19, 2010.
- [7] D. Malak and O. B. Akan, "Molecular communication nanonetworks inside human body," *Nano Communication Networks*, vol. 3, no. 1, pp. 19 – 35, 2012.
- [8] A. K. Geim and K. S. Novoselov, "The rise of graphene," *Nature Materials*, vol. 6, no. 3, pp. 183–191, 2007.
- [9] Y. L. Babikov, I. E. Gordon, S. N. Mikhailenko, L. S. Rothman, and S. A. Tashkun, "HITRAN on the Web – a new tool for HITRAN spectroscopic data manipulation," in *Proc. 12th Int. HITRAN Conf.*, (Université de Reims, Reims, France), 2012.
- [10] I. F. Akyildiz and J. M. Jornet, "The Internet of nano-things," *IEEE Wirel. Commun.*, vol. 17, pp. 58–63, dec 2010.
- [11] J. M. Jornet and I. F. Akyildiz, "Graphene-based nano-antennas for electromagnetic nanocommunications in the terahertz band," in *Proc. Fourth Eur. Conf. Antennas Propag.*, (Barcelona, Spain), pp. 1–5, IEEE, 2010.
- [12] S. Abadal, A. Cabellos-Aparicio, M. C. Lemme, M. Nemirovsky, and Others, "Graphene-enabled wireless communication for massive multicore architectures," *IEEE Commun. Mag.*, vol. 51, no. 11, pp. 137–143, 2013.
- [13] J. M. Jornet and I. F. Akyildiz, "Low-weight channel coding for interference mitigation in electromagnetic nanonetworks in the terahertz band," in *Commun. (ICC), 2011 IEEE Int. Conf.*, (Kyoto, Japan), pp. 1–6, IEEE, 2011.
- [14] J. M. Jornet and I. F. Akyildiz, "Channel Modeling and Capacity Analysis for Electromagnetic Wireless Nanonetworks in the Tera-

- hertz Band," *IEEE Trans. Wirel. Commun.*, vol. 10, pp. 3211–3221, oct 2011.
- [15] J. M. Jornet and I. F. Akyildiz, "Femtosecond-Long Pulse-Based Modulation for Terahertz Band Communication in Nanonetworks," *Commun. IEEE Trans.*, vol. 62, pp. 1742–1754, may 2014.
- [16] P. Boronin, V. Petrov, D. Moltchanov, Y. Koucheryavy, and J. M. Jornet, "Capacity and throughput analysis of nanoscale machine communication through transparency windows in the terahertz band," *Nano Commun. Netw.*, vol. 5, no. 3, pp. 72–82, 2014.
- [17] J. M. Jornet, J. C. Pujol, and J. S. Pareta, "Phlame: A physical layer aware MAC protocol for electromagnetic nanonetworks in the Terahertz band," *Nano Communication Networks*, vol. 3, no. 1, pp. 74–81, 2012.
- [18] P. Wang, J. M. Jornet, M. G. A. Malik, N. Akkari, and I. F. Akyildiz, "Energy and spectrum-aware MAC protocol for perpetual wireless nanosensor networks in the Terahertz Band," *Ad Hoc Networks*, vol. 11, no. 8, pp. 2541–2555, 2013.
- [19] S. Mohrehkesh and M. C. Weigle, "RIH-MAC: receiver-initiated harvesting-aware mac for nanonetworks," in *Proc. ACM First Annu. Int. Conf. Nanoscale Comput. Commun.*, pp. 1–9, ACM, 2014.
- [20] S. DOro, L. Galluccio, G. Morabito, and S. Palazzo, "A timing channel-based mac protocol for energy-efficient nanonetworks," *Nano Communication Networks*, vol. 6, no. 2, pp. 39 – 50, 2015. Pervasive and Ubiquitous Environment Interactions with Nano Things.
- [21] G. Piro, G. Boggia, and L. A. Grieco, "On the design of an energy-harvesting protocol stack for Body Area Nano-NETworks," *Nano Commun. Netw.*, 2014.

- [22] G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, "Nano-sim: Simulating electromagnetic-based nanonetworks in the network simulator 3," in *Proceedings of the 6th International ICST Conference on Simulation Tools and Techniques, SimuTools '13*, (ICST, Brussels, Belgium, Belgium), pp. 203–210, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2013.
- [23] A. Afsharinejad, A. Davy, B. Jennings, and S. Balasubramaniam, "GA-based frequency selection strategies for graphene-based nanocommunication networks," in *Proc. IEEE ICC*, (Sydney, Australia), pp. 3642–3647, 2014.
- [24] J. M. Jornet, "A joint energy harvesting and consumption model for self-powered nano-devices in nanonetworks," in *Proc. IEEE Int. Conf. Commun.*, (Ottawa, Canada), pp. 6151–6156, 2012.
- [25] J. M. Jornet, "Low-weight error-prevention codes for electromagnetic nanonetworks in the Terahertz Band," *Nano Communication Networks*, vol. 5, no. 1–2, pp. 35–44, 2014.
- [26] E. Zarepour, M. Hassan, C. T. Chou, and A. A. Adesina, "Frequency hopping strategies for improving terahertz sensor network performance over composition varying channels," in *Proc. IEEE WoW-MoM*, (Sydney), pp. 1–9, 2014.
- [27] I. F. Akyildiz, J. M. Jornet, and C. Han, "Terahertz band: Next frontier for wireless communications," *Phys. Commun.*, vol. 12, pp. 16–32, 2014.
- [28] L. Vangelista, A. Zanella, and M. Zorzi, "Long-range IoT technologies: The dawn of LoRa™," in *Futur. Access Enablers Ubiquitous Intell. Infrastructures*, pp. 51–58, Springer, 2015.

- [29] R. Ratasuk, N. Mangalvedhe, A. Ghosh, and B. Vejlgaard, "Narrow-band LTE-M system for M2M communication," in *Veh. Technol. Conf. (VTC Fall), 2014 IEEE 80th*, pp. 1–5, IEEE, 2014.
- [30] R. Ratasuk, B. Vejlgaard, N. Mangalvedhe, and A. Ghosh, "NB-IoT system for M2M communication," in *Wirel. Commun. Netw. Conf.*, pp. 1–5, IEEE, 2016.
- [31] D. C. Harrison, W. K. G. Seah, and R. Rayudu, "Rare event detection and propagation in wireless sensor networks," *ACM Comput. Surv.*, vol. 48, no. 4, p. 58, 2016.
- [32] C. Bisdikian, L. M. Kaplan, and M. B. Srivastava, "On the quality and value of information in sensor networks," *ACM Trans. Sens. Networks*, vol. 9, no. 4, p. 48, 2013.
- [33] I. F. Akyildiz, F. Brunetti, and C. Blázquez, "Nanonetworks: A new communication paradigm," *Computer Networks*, vol. 52, no. 12, pp. 2260–2279, 2008.
- [34] P. Lio and S. Balasubramaniam, "Opportunistic routing through conjugation in bacteria communication nanonetwork," *Nano Commun. Netw.*, vol. 3, no. 1, pp. 36–45, 2012.
- [35] B. Atakan, O. B. Akan, and S. Balasubramaniam, "Body area nanonetworks with molecular communications in nanomedicine," *IEEE Commun. Mag.*, vol. 50, no. 1, pp. 28–34, 2012.
- [36] M. c. Kuran, H. B. Yilmaz, T. Tugcu, and I. F. Akyildiz, "Interference effects on modulation techniques in diffusion based nanonetworks," *Nano Commun. Netw.*, vol. 3, no. 1, pp. 65–73, 2012.
- [37] S. Balasubramaniam and P. Lio, "Multi-hop conjugation based bacteria nanonetworks," *IEEE Trans. Nanobioscience*, vol. 12, no. 1, pp. 47–59, 2013.

- [38] V. Petrov, S. Balasubramaniam, Y. Koucheryavy, and M. Skurnik, "Forward and reverse coding for bacteria nanonetworks," in *Proc. 1st Int. Black Sea Conf. Commun. Netw.*, (Batumi, Georgia), pp. 74–78, IEEE, 2013.
- [39] S. Balasubramaniam, S. Ben-Yehuda, S. Pautot, A. Jesorka, Y. Koucheryavy, and Others, "A review of experimental opportunities for molecular communication," *Nano Commun. Netw.*, vol. 4, no. 2, pp. 43–52, 2013.
- [40] F. Walsh and S. Balasubramaniam, "Reliability and delay analysis of multihop virus-based nanonetworks," *Nanotechnology, IEEE Trans.*, vol. 12, no. 5, pp. 674–684, 2013.
- [41] M. Hasan, E. Hossain, S. Balasubramaniam, and Y. Koucheryavy, "Social Behavior in Bacterial Nanonetworks: Challenges and Opportunities," *IEEE Netw.*, 2015.
- [42] V. Petrov, D. Moltchanov, and Y. Koucheryavy, "Analytical Model for Link Reliability in Bacteria Nanonetworks," in *Proc. ACM First Annu. Int. Conf. Nanoscale Comput. Commun.*, pp. 1–7, ACM, 2014.
- [43] A. Noel, K. Cheung, and R. Schober, "Improving diffusion-based molecular communication with unanchored enzymes," in *Bio-Inspired Model. Network, Information, Comput. Syst.*, pp. 184–198, Springer, 2014.
- [44] B. D. Unluturk, M. S. Islam, S. Balasubramaniam, and S. Ivanov, "Towards concurrent data transmission: Exploiting plasmid diversity by bacterial conjugation," *IEEE Transactions on NanoBioscience*, vol. PP, no. 99, pp. 1–1, 2017.
- [45] B. D. Unluturk, S. Balasubramaniam, and I. F. Akyildiz, "The impact of social behavior on the attenuation and delay of bacterial nanonet-

- works," *IEEE Transactions on NanoBioscience*, vol. 15, pp. 959–969, Dec 2016.
- [46] Y. Chahibi, I. F. Akyildiz, S. Balasubramaniam, and Y. Koucheryavy, "Molecular communication modeling of antibody-mediated drug delivery systems," *IEEE Transactions on Biomedical Engineering*, vol. 62, pp. 1683–1695, July 2015.
- [47] H. Guo, P. Johari, J. M. Jornet, and Z. Sun, "Intra-body optical channel modeling for in vivo wireless nanosensor networks," *IEEE Transactions on NanoBioscience*, vol. 15, pp. 41–52, Jan 2016.
- [48] A. K. Geim, "Graphene: status and prospects," *Science (80-. )*, vol. 324, no. 5934, pp. 1530–1534, 2009.
- [49] C. Lee, X. Wei, J. W. Kysar, and J. Hone, "Measurement of the elastic properties and intrinsic strength of monolayer graphene," *Science (80-. )*, vol. 321, no. 5887, pp. 385–388, 2008.
- [50] M. A. Rafiee, J. Rafiee, I. Srivastava, Z. Wang, H. Song, Z.-Z. Yu, and N. Koratkar, "Fracture and fatigue in graphene nanocomposites," *small*, vol. 6, no. 2, pp. 179–183, 2010.
- [51] A. A. Balandin, "Thermal properties of graphene and nanostructured carbon materials," *Nat. Mater.*, vol. 10, no. 8, pp. 569–581, 2011.
- [52] G. Lalwani, A. M. Henslee, B. Farshid, L. Lin, F. K. Kasper, Y.-X. Qin, A. G. Mikos, and B. Sitharaman, "Two-dimensional nanostructure-reinforced biodegradable polymeric nanocomposites for bone tissue engineering," *Biomacromolecules*, vol. 14, no. 3, pp. 900–909, 2013.
- [53] S. Sayyar, E. Murray, B. C. Thompson, S. Gambhir, D. L. Officer, and G. G. Wallace, "Covalently linked biocompatible graphene/polycaprolactone composites for tissue engineering," *Carbon N. Y.*, vol. 52, pp. 296–304, 2013.

- [54] S. Goenka, V. Sant, and S. Sant, "Graphene-based nanomaterials for drug delivery and tissue engineering," *J. Control. Release*, vol. 173, pp. 75–88, 2014.
- [55] A. Olaofe and A. Szul, "THE APPLICATION OF CARBON NANOTUBE TECHNOLOGY TOWARDS BODY ARMOR FOR MILITARY USE," 2014.
- [56] Y. Hu, F. Li, D. Han, and L. Niu, "Graphene for Glucose, Dopamine, Ascorbic Acid, and Uric Acid Detection," in *Biocompatible Graphene Bioanal. Appl.*, pp. 57–79, Springer, 2015.
- [57] G. S. Kulkarni, K. Reddy, Z. Zhong, and X. Fan, "Graphene nano-electronic heterodyne sensor for rapid and sensitive vapour detection," *Nat. Commun.*, vol. 5, 2014.
- [58] T. Zhou, Z. Cheng, H. Zhang, M. Berre, L. Militaru, and F. Calmon, "Miniaturized tunable terahertz antenna based on graphene," *Microw. Opt. Technol. Lett.*, vol. 56, no. 8, pp. 1792–1794, 2014.
- [59] A. Afzali-Ardakani, P. Avouris, D. B. Farmer, Y.-M. Lin, and Y. Zhu, "Graphene transistors with self-aligned gates," 2014.
- [60] T. Romann, O. Oll, P. Pikma, K. Kirsimäe, and E. Lust, "4–10 V capacitors with graphene-based electrodes and ionic liquid electrolyte," *J. Power Sources*, vol. 280, pp. 606–611, 2015.
- [61] B. Sensale-Rodriguez, R. Yan, M. M. Kelly, T. Fang, K. Tahy, W. S. Hwang, D. Jena, L. Liu, and H. G. Xing, "Broadband graphene terahertz modulators enabled by intraband transitions," *Nat. Commun.*, vol. 3, p. 780, 2012.
- [62] A. Cabellos, I. Llátser, E. Alarcón, A. Hsu, and T. Palacios, "Use of THz Photoconductive Sources to Characterize Tunable Graphene RF Plasmonic Antennas," *arXiv Prepr. arXiv1401.6878*, 2014.

- [63] L. Zakrajsek, E. Einarsson, N. Thawdar, M. Medley, and J. M. Jornet, "Lithographically defined plasmonic graphene antennas for terahertz-band communication," *IEEE Antennas and Wireless Propagation Letters*, vol. 15, pp. 1553–1556, 2016.
- [64] M. Nafari and J. M. Jornet, "Modeling and performance analysis of metallic plasmonic nano-antennas for wireless optical communication in nanonetworks," *IEEE Access*, vol. 5, pp. 6389–6398, 2017.
- [65] L. Zakrajsek, E. Einarsson, N. Thawdar, M. Medley, and J. M. Jornet, "Design of graphene-based plasmonic nano-antenna arrays in the presence of mutual coupling," in *2017 11th European Conference on Antennas and Propagation (EUCAP)*, pp. 1381–1385, March 2017.
- [66] C. Stampfer, T. Helbling, D. Obergfell, B. Schöberle, M. K. Tripp, A. Jungen, S. Roth, V. M. Bright, and C. Hierold, "Fabrication of single-walled carbon-nanotube-based pressure sensors," *Nano Lett.*, vol. 6, no. 2, pp. 233–237, 2006.
- [67] A. A. Balandin, "Graphene-based gas and bio sensor with high sensitivity and selectivity," 2014.
- [68] J. M. Jornet and I. F. Akyildiz, "Graphene-based plasmonic nano-transceiver for terahertz band communication," in *Antennas Propag. (EuCAP), 2014 8th Eur. Conf.*, pp. 492–496, IEEE, 2014.
- [69] J. F. Federici, B. Schulkin, F. Huang, D. Gary, R. Barat, F. Oliveira, and D. Zimdars, "THz imaging and sensing for security applicationsexplosives, weapons and drugs," *Semicond. Sci. Technol.*, vol. 20, no. 7, p. S266, 2005.
- [70] Q. H. Abbasi, K. Yang, N. Chopra, J. M. Jornet, N. A. Abuali, K. A. Qaraqe, and A. Alomainy, "Nano-communication for biomedical applications: A review on the state-of-the-art from physical layers

- to novel networking concepts," *IEEE Access*, vol. 4, pp. 3920–3935, 2016.
- [71] I. F. Akyildiz, S. Nie, S.-C. Lin, and M. Chandrasekaran, "5g roadmap: 10 key enabling technologies," *Computer Networks*, vol. 106, pp. 17 – 48, 2016.
- [72] C. Wang, B. Lu, C. Lin, Q. Chen, L. Miao, X. Deng, and J. Zhang, "0.34-thz wireless link based on high-order modulation for future wireless local area network applications," *IEEE Transactions on Terahertz Science and Technology*, vol. 4, pp. 75–85, Jan 2014.
- [73] T. Narytnik, A. Amro, M. Ilchenko, V. Kalinin, and O. Turabi, "Sub-terahertz low power uwb communication link for wpan," *Network and Complex Systems*, vol. 2, no. 4, pp. 45–49, 2012.
- [74] D. L. Woolard, J. O. Jensen, and R. J. Hwu, *Terahertz science and technology for military and security applications*, vol. 46. world scientific, 2007.
- [75] H. Elayan, R. M. Shubair, and J. M. Jornet, "Bio-electromagnetic thz propagation modeling for in-vivo wireless nanosensor networks," in *2017 11th European Conference on Antennas and Propagation (EU-CAP)*, pp. 426–430, March 2017.
- [76] M. Walther, B. M. Fischer, A. Ortner, A. Bitzer, A. Thoman, and H. Helm, "Chemical sensing and imaging with pulsed terahertz radiation," *Analytical and Bioanalytical Chemistry*, vol. 397, pp. 1009–1017, Jun 2010.
- [77] M. Nafari, L. Feng, and J. M. Jornet, "On-chip wireless optical channel modeling for massive multi-core computing architectures," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, March 2017.

- [78] "Ieee 802.15 wpan task group 3d 100 gbit/s wireless (tg 3d (100g))[online], [http://www.ieee802.org/15/pub/index\\_TG3d.html](http://www.ieee802.org/15/pub/index_TG3d.html),"
- [79] C. Han, I. F. Akyildiz, and W. H. Gerstacker, "Timing acquisition for pulse-based wireless systems in the terahertz band," in *Proceedings of the Second Annual International Conference on Nanoscale Computing and Communication*, NANOCOM' 15, (New York, NY, USA), pp. 7:1–7:6, ACM, 2015.
- [80] J. F. Federici, J. Ma, and L. Moeller, "Review of weather impact on outdoor terahertz wireless communication links," *Nano Communication Networks*, vol. 10, pp. 13 – 26, 2016. Terahertz Communications.
- [81] C. R. Yonzon, D. A. Stuart, X. Zhang, A. D. McFarland, C. L. Haynes, and R. P. V. Duyne, "Towards advanced chemical and biological nanosensorsan overview," *Talanta*, vol. 67, no. 3, pp. 438 – 448, 2005. Nanoscience and Nanotechnology.
- [82] C. Li, E. T. Thostenson, and T.-W. Chou, "Sensors and actuators based on carbon nanotubes and their composites: A review," *Composites Science and Technology*, vol. 68, no. 6, pp. 1227 – 1249, 2008.
- [83] F. Vullum and D. Teeters, "Investigation of lithium battery nanoelectrode arrays and their component nanobatteries," *Journal of Power Sources*, vol. 146, no. 1, pp. 804 – 808, 2005. Selected papers presented at the 12th International Meeting on Lithium Batteries.
- [84] Z. L. Wang, "Towards self-powered nanosystems: From nanogenerators to nanopiezotronics," *Advanced Functional Materials*, vol. 18, no. 22, pp. 3553–3567, 2008.
- [85] Y. Ouyang, Y. Yoon, J. K. Fodor, and J. Guo, "Comparison of performance limits for carbon nanoribbon and carbon nanotube transistors," *Applied Physics Letters*, vol. 89, no. 20, p. 203107, 2006.

- [86] K. Y. Tan, K. W. Chan, M. Mttinen, A. Morello, C. Yang, J. v. Donkelaar, A. Alves, J.-M. Pirkkalainen, D. N. Jamieson, R. G. Clark, and A. S. Dzurak, "Transport spectroscopy of single phosphorus donors in a silicon nanoscale transistor," *Nano Letters*, vol. 10, no. 1, pp. 11–15, 2010. PMID: 19950969.
- [87] R. Bennewitz, J. N. Crain, A. Kirakosian, J.-L. Lin, J. L. McChesney, D. Y. Petrovykh, and F. J. Himpsel, "Atomic scale memory at a silicon surface," *Nanotechnology*, vol. 13, no. 4, p. 499, 2002.
- [88] S. S. P. Parkin, M. Hayashi, and L. Thomas, "Magnetic domain-wall racetrack memory," *Science*, vol. 320, no. 5873, pp. 190–194, 2008.
- [89] J. M. Jornet and I. F. Akyildiz, "Graphene-based plasmonic nanotransceiver for terahertz band communication," in *The 8th European Conference on Antennas and Propagation (EuCAP 2014)*, pp. 492–496, April 2014.
- [90] C. Rutherglen and P. Burke, "Carbon nanotube radio," *Nano Letters*, vol. 7, no. 11, pp. 3296–3299, 2007. PMID: 17941677.
- [91] G. Piro, G. Boggia, and L. A. Grieco, "On the design of an energy-harvesting protocol stack for body area nano-networks," *Nano Communication Networks*, vol. 6, no. 2, pp. 74 – 84, 2015. Pervasive and Ubiquitous Environment Interactions with Nano Things.
- [92] L. Hou, S. Zhao, X. Xiong, K. Zheng, P. Chatzimisios, M. S. Hossain, and W. Xiang, "Internet of Things Cloud: Architecture and Implementation," *IEEE Commun. Mag.*, vol. 54, pp. 32–39, dec 2016.
- [93] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An Information Framework for Creating a Smart City Through Internet of Things," *IEEE Internet Things J.*, vol. 1, pp. 112–121, apr 2014.

- [94] G. Maduenno, C. Stefanovic, and P. Popovski, "Reliable and Efficient Access for Alarm-initiated and Regular M2M Traffic in IEEE 802.11ah Systems," *IEEE Internet Things J.*, vol. 3, pp. 673–682, oct 2015.
- [95] E. Piri and J. Pinola, "Performance of LTE uplink for IoT backhaul," in *Proc. 13th IEEE Annu. Consum. Commun. Netw. Conf.*, (Las Vegas, NV, USA), pp. 6–11, 2016.
- [96] T. P. C. de Andrade, C. A. Astudillo, and N. L. S. da Fonseca, "Allocation of Control Resources for Machine-to-Machine and Human-to-Human Communications Over LTE/LTE-A Networks," *IEEE Internet Things J.*, vol. 3, pp. 366–377, jun 2016.
- [97] M. R. Palattella, M. Dohler, A. Grieco, G. Rizzo, J. Torsner, T. Engel, and L. Ladid, "Internet of Things in the 5G Era: Enablers, Architecture, and Business Models," *IEEE J. Sel. Areas Commun.*, vol. 34, pp. 510–527, mar 2016.
- [98] G. Margelis, R. Piechocki, D. Kaleshi, and P. Thomas, "Low Throughput Networks for the IoT: Lessons learned from industrial implementations," in *Proc. IEEE 2nd World Forum Internet Things*, (Milan, Italy), pp. 181–186, IEEE, 2015.
- [99] K. Zheng, S. Zhao, Z. Yang, X. Xiong, and W. Xiang, "Design and Implementation of LPWA-Based Air Quality Monitoring System," *IEEE Access*, vol. 4, pp. 3238–3245, 2016.
- [100] D. Bercovich, L. M. Contreras, Y. Haddad, A. Adam, and C. J. Bernardos, "Software-Defined Wireless Transport Networks for Flexible Mobile Backhaul in 5G Systems," *Mob. Netw. Appl.*, vol. 20, pp. 793–801, dec 2015.
- [101] J. M. Dubach, D. I. Harjes, and H. A. Clark, "Fluorescent ion-selective nanosensors for intracellular analysis with improved life-

- time and size," *Nano Letters*, vol. 7, no. 6, pp. 1827–1831, 2007. PMID: 17497824.
- [102] Z. S. H. Guo and J. M. Jornet, "A Cooperative Raman Spectrum Reconstruction Platform for Real-time In-vivo Nano-biosensing," in *Proc. INFOCOM Work. Wirel. Commun. Netw. Extrem. Environ.*, (Atlanta, GA, USA), pp. 1–6, may 2017.
- [103] E. Zarepour, M. Hassan, C. T. Chou, and M. E. Warkiani, "Characterizing terahertz channels for monitoring human lungs with wireless nanosensor networks," *Nano Commun. Netw.*, vol. 9, pp. 43–57, 2016.
- [104] N. Islam and S. Misra, "'catch the pendulum': The problem of asymmetric data delivery in electromagnetic nanonetworks," *IEEE Transactions on NanoBioscience*, vol. 15, pp. 576–584, Sept 2016.
- [105] I. Renato, V. Loscri, S. Pizzi, R. Tarparelli, and A. M. Vegni, "Electromagnetic Nanonetworks for Sensing and Drug Delivery," in *Modeling, Methodologies and Tools for Molecular and Nano-scale Communications* (J. Suzuki, T. Nakano, and M. J. Moore, eds.), Springer, Dec. 2014.
- [106] A. Afsharinejad, A. Davy, B. Jennings, and C. Brennan, "An initial path-loss model within vegetation in the thz band," in *2015 9th European Conference on Antennas and Propagation (EuCAP)*, pp. 1–5, May 2015.
- [107] A. Afsharinejad, A. Davy, B. Jennings, S. Rasmann, and C. Brennan, "A path-loss model incorporating shadowing for thz band propagation in vegetation," in *2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, Dec 2015.
- [108] A. Afsharinejad, A. Davy, and M. Naftaly, "Variability of terahertz transmission measured in live plant leaves," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, pp. 636–638, May 2017.

- [109] A. Afsharinejad, A. Davy, B. Jennings, and C. Brennan, "Performance analysis of plant monitoring nanosensor networks at thz frequencies," *IEEE Internet of Things Journal*, vol. 3, pp. 59–69, Feb 2016.
- [110] A. Afsharinejad, A. Davy, and B. Jennings, "Dynamic channel allocation in electromagnetic nanonetworks for high resolution monitoring of plants," *Nano Communication Networks*, vol. 7, pp. 2 – 16, 2016.
- [111] C. M. Pieterse and M. Dicke, "Plant interactions with microbes and insects: from molecular mechanisms to ecology," *Trends in Plant Science*, vol. 12, no. 12, pp. 564 – 569, 2007.
- [112] R. Crowder, "Toward robots that can sense texture by touch," *Science*, vol. 312, no. 5779, pp. 1478–1479, 2006.
- [113] E. Zarepour, M. Hassan, C. T. Chou, and A. A. Adesina, "Nano-scale sensor networks for chemical catalysis," in *2013 13th IEEE International Conference on Nanotechnology (IEEE-NANO 2013)*, pp. 61–66, Aug 2013.
- [114] E. Zarepour, M. Hassan, C. T. Chou, and A. A. Adesina, "Remote detection of chemical reactions using nanoscale terahertz communication powered by pyroelectric energy harvesting," in *Proceedings of the Second Annual International Conference on Nanoscale Computing and Communication, NANOCOM' 15*, (New York, NY, USA), pp. 8:1–8:6, ACM, 2015.
- [115] S. Abadal, C. Liaskos, A. Tsioliariidou, S. Ioannidis, A. Pitsillides, J. Sole-Pareta, E. Alarcon, and A. Cabellos-Aparicio, "Computing and communications for the software-defined metamaterial paradigm: A context analysis," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2017.

- [116] H. Zhang, E. Bilotti, and T. Peijs, "The use of carbon nanotubes for damage sensing and structural health monitoring in laminated composites: a review," *Nanocomposites*, vol. 1, no. 4, pp. 167–184, 2015.
- [117] A. Afsharinejad, A. Davy, and B. Jennings, "Frequency selection strategies under varying moisture levels in wireless nanonetworks," in *Proceedings of ACM The First Annual International Conference on Nanoscale Computing and Communication, NANOCOM' 14*, (New York, NY, USA), pp. 23:1–23:9, ACM, 2007.
- [118] J. C. Pujol, J. M. Jornet, and J. S. Pareta, "Phlame: A physical layer aware mac protocol for electromagnetic nanonetworks," in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 431–436, April 2011.
- [119] A. P. Shrestha, S. J. Yoo, H. J. Choi, and K. S. Kwak, "Enhanced rate division multiple access for electromagnetic nanonetworks," *IEEE Sensors Journal*, vol. 16, pp. 7287–7296, Oct 2016.
- [120] S. Mohrehkesh, M. C. Weigle, and S. K. Das, "Drih-mac: A distributed receiver-initiated harvesting-aware mac for nanonetworks," *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, vol. 1, pp. 97–110, March 2015.
- [121] Q. Xia, Z. Hossain, M. Medley, and J. M. Jornet, "A link-layer synchronization and medium access control protocol for terahertz-band communication networks," in *Glob. Commun. Conf. (GLOBECOM), 2015 IEEE*, pp. 1–7, IEEE, 2015.
- [122] N. Akkari, P. Wang, J. Jornet, E. Fadel, L. Elrefaei, A. Malik, S. Almasri, and F. A. Ian, "Distributed Timely-Throughput Optimal Scheduling for the Internet of Nano-Things," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1202–1212, 2016.

- [123] M. Pierobon, J. M. Jornet, N. Akkari, S. Almasri, and I. F. Akyildiz, "A routing framework for energy harvesting wireless nanosensor networks in the Terahertz Band," *Wirel. Networks*, vol. 20, no. 5, pp. 1–15, 2014.
- [124] A. Tsioliaridou, C. Liaskos, S. Ioannidis, and A. Pitsillides, "Corona: A coordinate and routing system for nanonetworks," in *Proceedings of the Second Annual International Conference on Nanoscale Computing and Communication, NANOCOM' 15*, (New York, NY, USA), pp. 18:1–18:6, ACM, 2015.
- [125] A. Tsioliaridou, C. Liaskos, L. Pachis, S. Ioannidis, and A. Pitsillides, "N3: Addressing and routing in 3d nanonetworks," in *2016 23rd International Conference on Telecommunications (ICT)*, pp. 1–6, May 2016.
- [126] A. Tsioliaridou, C. Liaskos, E. Dedu, and S. Ioannidis, "Stateless linear-path routing for 3d nanonetworks," in *Proceedings of the 3rd ACM International Conference on Nanoscale Computing and Communication, NANOCOM'16*, (New York, NY, USA), pp. 28:1–28:6, ACM, 2016.
- [127] A. Tsioliaridou, C. Liaskos, E. Dedu, and S. Ioannidis, "Packet routing in 3d nanonetworks: A lightweight, linear-path scheme," *Nano Communication Networks*, vol. 12, no. Supplement C, pp. 63–71, 2017.
- [128] C. Liaskos and A. Tsioliaridou, "A promise of realizable, ultra-scalable communications at nano-scale: a multi-modal nano-machine architecture," *IEEE Transactions on Computers*, vol. 64, pp. 1282–1295, May 2015.
- [129] A. Tsioliaridou, C. Liaskos, S. Ioannidis, and A. Pitsillides, "Lightweight, self-tuning data dissemination for dense nanonetworks," *Nano Communication Networks*, vol. 8, no. Supplement C, pp. 2–15, 2016. Electromagnetic Communication in Nano-scale.

- [130] C. Liaskos, A. Tsioliariidou, S. Ioannidis, N. Kantartzis, and A. Pitsillides, "A deployable routing system for nanonetworks," in *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6, May 2016.
- [131] T. A. A. Alsbou, M. Hammoudeh, Z. Bandar, and A. Nisbet, "An overview and classification of approaches to information extraction in wireless sensor networks," in *Proc. 5th Int. Conf. Sens. Technol. Appl.*, p. 255, 2011.
- [132] D. C. Harrison, D. Burmester, W. K. G. Seah, and R. Rayudu, "Busting myths of energy models for wireless sensor networks," *Electron. Lett.*, vol. 52, no. 16, pp. 1412–1414, 2016.
- [133] Z. Zhang, M. Ma, and Y. Yang, "Energy-efficient multihop polling in clusters of two-layered heterogeneous sensor networks," *IEEE Trans. Comput.*, vol. 57, no. 2, pp. 231–245, 2008.
- [134] Z. Liang and D. Zhao, "A priority-based polling scheme for clustering wireless sensor network MAC protocols," in *Proc 1st Int. Conf. Inf. Sci. Eng.*, (Nanjing, China), pp. 2488–2491, 2009.
- [135] W. K. G. Seah, Z. A. Eu, and H.-P. Tan, "Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP)-Survey and challenges," in *Proc. 1st Int. Conf. Wirel. Commun. Veh. Technol. Inf. Theory Aerosp. Electron. Syst. Technol. (Wireless VITAE)*, (Aalborg, Denmark), pp. 1–5, 2009.
- [136] Z. A. Eu, H.-P. Tan, and W. K. G. Seah, "Design and performance analysis of MAC schemes for wireless sensor networks powered by ambient energy harvesting," *Ad Hoc Networks*, vol. 9, no. 3, pp. 300–323, 2011.
- [137] C. Fujii and W. K. G. Seah, "Multi-tier probabilistic polling in wireless sensor networks powered by energy harvesting," in *Proc 7th Int.*

- Conf. Intell. Sensors, Sens. Networks Inf. Process.*, (Adelaide, SA, Australia), pp. 383–388, 2011.
- [138] M. Kunikawa, H. Yomo, K. Abe, and T. Ito, “A Fair Polling Scheme for Energy Harvesting Wireless Sensor Networks,” in *Proc. IEEE 81st Veh. Technol. Conf. (VTC Spring)*, (Glasgow, Scotland), pp. 1–5, 2015.
- [139] Y. Gao, D.-M. Chiu, and J. C. S. Lui, “Determining the End-to-end Throughput Capacity in Multi-hop Networks: Methodology and Applications,” in *Proc. ACM SIGMETRICS*, (Saint Malo, France), pp. 39–50, 2006.
- [140] H. Zhao, E. Garcia-Palacios, A. Song, and J. Wei, “Calculating End-to-End Throughput Capacity in Wireless Networks with Consideration of Hidden Nodes and Multi-Rate Terminals,” in *Proc. IEEE VTC, 2011 IEEE 73rd*, pp. 1–5, may 2011.
- [141] S. Luo, Y. Sun, and Y. Ji, “Data collection for time-critical applications in the low-duty-cycle wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 11, no. 8, p. 931913, 2015.
- [142] M. Zafer and E. Modiano, “Delay-Constrained Energy Efficient Data Transmission over a Wireless Fading Channel,” in *2007 Inf. Theory Appl. Work.*, pp. 289–298, jan 2007.
- [143] Y. Chen, P. S. Anwar, L. Huang, and M. Asvial, “Characterizing nanoscale transient communication,” *IEEE Transactions on NanoBioscience*, vol. 15, pp. 218–229, April 2016.
- [144] K. Zheng, F. Hu, W. Wang, W. Xiang, and M. Dohler, “Radio resource allocation in LTE-advanced cellular networks with M2M communications,” *IEEE Commun. Mag.*, vol. 50, no. 7, pp. 184–192, 2012.
- [145] G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, “Nano-Sim: simulating electromagnetic-based nanonetworks in the network simula-

- tor 3," in *Proc. 6th Int. ICST Conf. Simul. Tools Tech.*, (Cannes, French Riviera), pp. 203–210, 2013.
- [146] C. G. C.HARPHAM and P. JONES, "Betwixt built environment: Weather scenarios for investigation of impacts and extremes, BETWIXT technical briefing note 7 version 1," tech. rep., Climatic Research Unit, University of East Anglia, June 2006.
- [147] N. Baccour, A. Koubâa, L. Mottola, M. A. Zúñiga, H. Youssef, C. A. Boano, and M. Alves, "Radio link quality estimation in wireless sensor networks: a survey," *ACM Trans. Sens. Networks*, vol. 8, no. 4, p. 34, 2012.
- [148] S. DOro, L. Galluccio, G. Morabito, and S. Palazzo, "A timing channel-based mac protocol for energy-efficient nanonetworks," *Nano Communication Networks*, vol. 6, no. 2, pp. 39 – 50, 2015. Pervasive and Ubiquitous Environment Interactions with Nano Things.
- [149] F. Peper, K. Leibnitz, J. n. Teramae, T. Shimokawa, and N. Wakamiya, "Low-Complexity Nanosensor Networking Through Spike-Encoded Signaling," *IEEE Internet Things J.*, vol. 3, pp. 49–58, feb 2016.
- [150] Q. H. Abbasi, K. Yang, N. Chopra, J. M. Jornet, N. A. Abuali, K. A. Qaraqe, and A. Alomainy, "Nano-Communication for Biomedical Applications: A Review on the State-of-the-Art From Physical Layers to Novel Networking Concepts," *IEEE Access*, vol. 4, pp. 3920–3935, 2016.