ABSTRACT

PAMBUDI, SIGIT ARYO. Modeling and Evaluation of Cyber-Physical Threats in Emerging Interdependent Networks. (Under the direction of Wenye Wang and Xiaogang Wang.)

The proliferation of computer networking technologies have enabled many emerging paradigms such as mobile social networking (MSN), smart grid, and Internet-of-things (IoT). In these paradigms, the computer networks not only support, but also tightly interact with other cyber (i.e., social networks) and/or physical (i.e., power and infrastructure networks) entities, forming an *interdependent cyber and physical networks.* Unfortunately, the interdependence relation in cyber-physical networks poses a major challenge in analyzing the networks' performance. Moreover, various man-made and natural threats may induce faults, which due to the interdependence can spread to a larger part of the cyber-physical networks, causing a so-called *cascade-of-failures.* Since billions of people will depend on cyber-physical networks, performance degradation—especially under threat-induced cascade-of-failures—will have a profound impact and remains as an open yet interesting problem.

In this dissertation, we aim to understand the performance of cyber-physical networks and its robustness against man-made and natural faults. Because it is not practical to provide a unified framework for all types of cyber-physical networks, which have their own features and objectives, we adopt a systematic, step-by-step approach to model and evaluate the networks' performance and robustness. Specifically, we focus on four emerging paradigms: D2D-based IoT, IoT-applied dynamic spectrum access (DSA) network, D2D-based MSN, and smart grid.

In particular, we first consider the *intra-network* interdependence in the communication network by studying the delay of reaching a point-of-access in D2D-based IoT applications. We propose a packet mobility model to capture the progress of data packets toward the point-of-access, which is used to derive the upper and lower bounds of access delay. Then, we propose a packet shedding scheme that reduces the total transmit power while guaranteeing that packets can be delivered within a pre-determined access delay. Second, we study how to quickly establish a common control channel between a pair of DSA nodes with limited channel hopping capability. We introduce graph-based channel hopping schemes with quick rendezvous and show that the proposed schemes achieve satisfactory rendezvous rate under an indoor mobile environment. In addition, we consider the *inter-network* interdependence between the cyber and physical networks by analyzing the impact of initial node dropouts to the network-wide and end-users' resilience of D2D-based MSNs. Finally, we characterize the spatial and temporal impacts of edge disconnections in generic cyber-physical networks. The works in this dissertation not only expand our knowledge on the performance of cyber-physical networks, but also provide instrumental guidelines to the design of robust interdependent networks.

Modeling and Evaluation of Cyber-Physical Threats in Emerging Interdependent Networks

by
Sigit Aryo Pambudi

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Computer Engineering

Raleigh, North Carolina

2017

APPROVED BY:

George Rouskas	Min Kang
Wenye Wang Co-chair of Advisory Committee	Xiaogang Wang Co-chair of Advisory Committee

DEDICATION

To my mom, Lily Endang Setyorini, and my dad, Suhardjito.
To my wife, Ilma Andayana, for her endless wisdom.
To my little angle, Adrian Pambudi.

BIOGRAPHY

Sigit Aryo Pambudi earned his B.S. degree in Electrical Engineering from Bandung Institute of Technology in 2009 and his M.S. degree from Yonsei University in 2011. In Auguts 2012, he started his Ph.D. study at North Carolina State University in the Department of Electrical and Computer Engineering. His research is focused on modeling the performance and reliability of cyber-physical systems, as well as designing novel approaches for defending complex, interdependent networks against attacks.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my advisor, Dr. Wenye Wang, and my co-advisor, Dr. Xiaogang "Cliff" Wang. I am very blessed to have the opportunity to work closely with Dr. Wenye Wang and learn from her research passion, wisdom, and much more, the attitude toward life. I sincerely appreciate Dr. Cliff Wang's suggestions, encouragement, and guidance during my Ph.D. study. I believe that the knowledge, skill, and persistent they have imparted upon me span far beyond this dissertation.

I would also like to thank my committee members, Dr. George Rouskas and Dr. Min Kang, for their valuable feedback and comments, which have significantly improve the quality of my research and dissertation. I am also tankful to all the professors at NC State for sharing their knowledge in and out of classes, which allows me to be prepared for my future career.

My gratitude also goes to my fellow lab mates for their help during my Ph.D. study: Zhuo Lu, Yujin Li, Jianhua Zhang, Mohit Shah, Huan Luo, Mingkui Wei, Wei Fu, Jie Wang, Rui Zou, Jiexiao Yu, Yan Yu, and Yali Wang.

Finally, I would like to give my special thanks to my wife Ilma; my son Adrian; my parents, Jimmy and Rini; my sister, Arini; and my parents-in-law, Berlian and Roza, for their encouragement and tireless support during my Ph.D. study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Motivation

The everlasting development of computer networking technologies and their adoption into daily activities have brought tremendous changes to people's lives. For example, 4G cellular standard, which has gained billions of subscribers world-wide, provides not only phone call and text messaging services but also high-speed mobile data access for web surfing, video conference, navigation, gaming, and virtual reality. On the other hand, WiFi standard have become more flexible and versatile, providing ubiquitous and high-speed wireless access to the Internet. More recently, ad-hoc wireless networking standards such as Bluetooth peer-to-peer, WiFi Direct, and device-to-device (D2D) [9, 55, 150] have provided a low-cost, low-power means for exchanging data in areas without cellular nor WiFi coverage. Such an ever-expanding versatility provided by modern computer networks has further enabled many emerging networking paradigms. For example, wireless and wire-line technologies have been deployed in smart grid [151, 28, 62, 127, 148] to help next-generation power systems fulfill the goal of Energy Internet, which features efficient and intelligent management of both power and information flows. Moreover, the ability of modern smart phones and tablets to form short-range wireless ad-hoc networks have been exploited to enable D2D-based mobile social networking (MSN) services such as FireChat and Jott [2, 37]. In a much larger scale, various networking technologies have also been used to connect smart everyday "things" to the Internet, forming an Internet-of-Things (IoT) paradigm [22, 8, 10]. These emerging paradigms have brought a wide array of economic, educational, political, accessibility, and health benefits to the society [12].

To provide a multitude of benefits to the society, emerging computer networking paradigms need to bring together many different but correlated entities, which results in two types of relations as follows. The first relation stems from the fact that the computer network itself consists of both physical entities—such as device hardware, packet buffers, node mobility pat-

tern, and/or wired/wireless communication medium—and virtual entities—in form of packet routing algorithm, resource sharing and contention mechanism, device operating system, and so on—working in unison to provide networking services. To this end, the operation of one entity depends on the function provided by others. For example, node mobility pattern may determine which routing protocol to implement, further impacting the necessary processing power and device size, which in turn can limit the device's mobility. Notice that the dependence between the mobility pattern and routing protocol is reciprocal and occurs within the communication network itself; thus we refer such a relation as an *intra-network interdependence*. Due to intra-network interdependence, the properties and dynamics of even a single entity cannot be studied in a straightforward manner. As a result, the analysis of emerging networks' performance, which consists of the properties of multiple entities, becomes non-trivial.

In emerging networking paradigms, the computer network typically needs to interact with other cyber and/or physical infrastructure networks, forming a so-called *cyber-physical networks.* For example, wired and/or wireless networks may interact with a physical power distribution network in smart grid, while the wireless network needs to support MSN service as the cyber network in a D2D-based MSN. In the former smart grid example, the nodes in the computer network depends on the electricity supplied by the power grid, while smart power nodes require the control information from the computer network. This results in the second type of network relationship: *inter-network interdependence.*

Inter-network and intra-network interdependencies become an important primitive especially when the emerging cyber-physical networks are vulnerable to man-made and natural disasters. For example, in D2D-based networking services, the open nature of D2D wirelss channels enables malicious adversaries to launch *spoofing attack* to steal data, spread malicious software (malware), or bypass security [161, 167] and/or initiate jamming attack to prevent legitimate users from accessing network resources [103, 161, 132]. On the other hand, natural phenomena like hurricane and thunderstorm can topple down trees onto active transmission lines in smart grid, hindering electricity delivery to the customers [160, 138, 28]. In the presence of interdependence, an initial fault at the first network (i.e., an electrical line failure in the power grid) may trigger failures at the second network (e.g. a control node disconnection in the communication network), which further induces additional breakdowns to the first network. This vicious cycle may repeat itself over time, leading to a so-called *cascade of failures* [150, 38, 114, 84]. As a result, a small initial threat in interdependent networks may develop into large-scale failures that will diminish the networks' availability. Without availability, the cyber-physical networks are just a collection of separated nodes without being able to provide networking service.

In order to achieve efficient and reliable networking, it is important to address the problem of understanding emerging cyber-physical networks' performance and robustness under threats.

However, such a knowledge is still largely missing in the literature. Therefore, in this dissertation, we are committed to study the performance of emerging cyber-physical systems as well as their robustness under man-made and natural threats.

Before studying the performance and robustness of emerging cyber-physical networks, we notice that different applications and services demand inherently distinct features and objectives. For example, service-based paradigms such as smart grid are designed to provide utility and resources to as many users as possible. In contrast, other paradigms such as IoT for disaster recovery are time-critical and designed to successfully deliver information messages within a tight deadline. These examples indicate that using a single, unified framework to measure the performance and robustness of cyber-physical networks is impractical. As a result, we apply a systematic, step-by-step approach for identifying, modeling, and evaluating the performance metrics and robustness criteria for emerging cyber-physical networks.

We start our study by excluding the inter-network interdependence and considering the simplest form of emerging cyber-physical networks: a mobile network with intra-network interdependence. Specifically, we consider and IoT paradigm, which is predicted to gain popularity very rapidly and have up to 16 billion devices by 2021 [8]. IoT paradigm hinges upon three main layers [78]: (i) hardware, made up of sensors, actuators, and embedded communication modules, (ii) middleware, i.e., on-demand storage and computing tools for data analysis, and (iii) presentation, consisting novel visual and interpretation tools. The seamless operation of these three layers relies heavily in the critical first *stage*, in which sensing data obtained by the "things" must be delivered to and aggregated at the middleware in forms of *cloudlets* [63]. Unfortunately, most cloudlet-based IoT studies assume that the network area is sufficiently covered by the cloudlets and the device-to-cloudlet links (i.e., the first stage) are stable [122, 63], which simply does not hold in challenging environments such as wide-area networks. In a challenging networking environment, IoT devices are mobile and highly power constrained, the cloudlets are sparsely-located, and/or hilly terrain limits the wireless signal's range [22, 78, 112, 86]. In these environments, every "thing" periodically senses their surrounding, encapsulates the sensing data into a packet ϵ, and then sends the packet to its neighbors. These neighboring nodes can further store ϵ to their internal buffers, carry it as they move in the network, forward ϵ to any nearby nodes, and then repeat this *store-carry-forward* (SCF) process [174] until ϵ can be delivered to a cloudlet. Unlike the existing SCF solutions, IoT application differs in the sense that the destination cloudlet is fixed, instead of mobile as in mobile ad-hoc networks (MANETs) [67, 72], and the supported application—such as search and rescue in disaster recovery—is time critical, instead of delay-tolerant [91, 73]. Motivated by the lack of suitable studies, we ask, *"What is the cloudlet access delay for IoT-based applications in challenging environment, and how to support time-critical applications while minimizing power consumption?"*

Upon studying IoT-based networks, we notice that their vast amount of nodes will require a

more efficient way of accessing the wireless spectrum. To this end, software-defined radio (SDR) technology enables IoT nodes to intelligently adjust their carrier frequency and efficiently access unoccupied wireless channels, which results in a so-called dynamic spectrum access (DSA) paradigm [173]. In DSA networks, one of the most challenging problem is for a pair of nodes to *rendezvous* [165], that is, to hop to a common channel for initiating a common control channel establishment process. Many researchers have attempted to achieve fast rendezvous by letting DSA nodes to perform channel hopping (CH) according to orthogonal sequences [165, 159, 31, 45, 166, 24]. However, the existing studies simply neglect the fact that practical DSA nodes have limited CH capability, which can be attributed due to their bounded channel switching time and restricted channel sensing capability. For example, modern Ettus E310 USRP SDR only has a digital bandwidth of 56MHz [26] and hopping to an arbitrary channel spaced more than 56MHz apart from the existing channel requires more than 30ms of switching time, which is way beyond the required 10ms CH interval [65]. Motivated by the lack of suitable schemes, we ask, *"How to design a CH scheme with fast rendezvous for DSA nodes with limited CH capability?"*

As the next step of our study, we consider emerging cyber-physical networks, but with inter-network interdependence. Recently, a new paradigm called D2D-based MSN that uses wireless mesh networking technologies such as Bluetooth peer-to-peer and WiFi Direct to provide MSN service to proximate smart phone users has been proposed commercially [2, 37]. In terms of its core functionality, D2D-based MSN consists of two different networks: a physical computer network in form of mobile devices and D2D wireless links among them, and a virtual social network (MSN) that captures the friendship between users. The social links (friendships) provide the possible endpoints of social contents, which are sent as packets using multi-hop D2D communications. A social user cannot exchange social contents with his/her friends without having its mobile device connected to the communication network, while a user who cannot find any friend will deactivate its mobile device and be dropped out from the communication network. As a result, D2D-based MSN can be viewed as a collection of cyber-physical networks with inter-network interdependence. Unfortunately, the open and shared nature of the wireless medium exposes the physical nodes to proximity-based malware infection, while the social users that become inactive may decide to abandon and be dropped out of the MSN. Due to the inter-network interdependence, a small dropout can develop into a system-wide disconnections and unavailability. Without availability, D2D-based MSN cannot be used to disseminate information, which may range from paid advertisements to disaster alerts [3]. Although there are studies regarding whether a network-wide cascade-of-failures occur in interdependent networks [38, 85], little has been known regarding how many nodes remain functional after a cascade. Moreover, existing researches have not considered the impact of cascades to the users' quality-of-experience. Hence we are interested in addressing both problems by analyzing the spatial

and temporal impacts of dropout-induced contractions in D2D-based MSN.

In addition to initial node dropouts, cyber-physical networks with inter-network interdependence may also be susceptible to edge disconnections. For example, edges in smart grid are installed as overhead power lines that are exposed to the elements and easily toppled down by disasters like tornadoes and hurricanes. On the other hand, a pair of wireless devices in D2D-based MSN may be obstructed by many objects in-between that cause severe attenuation and intermittent link disconnections [153]. Although several scientists have attempted to study the impacts of edge disconnections to network connectivity in stand-alone power grid [160, 172], the impact of edge disconnections in cyber-physical networks with interdependence is still an open problem. Therefore, this open problem motivates us to evaluate the impact of edge disconnections in interdependent cyber-physical networks.

In summary, our systematic, step-by-step study regarding intra-network and inter-network interdependence in cyber-physical networks not only expands our knowledge on the performance of cyber-physical networks, but also provides instrumental guidelines to the design of robust interdependent networks.

1.2 Research Problems and Contribution

Given the four open research problems identified above, we next elaborate on how we investigate these individual problems and also highlight our contributions in this dissertation.

1.2.1 Fast and Efficient Access of Cloudlet for IoT-Based Applications in Hostile Environments

To measure the amount of time to access a cloudlet, we follow the progress of packets' location in relation to the destination cloudlet, which is captured by a novel packet mobility model. Then, we divide the packet mobility into three generic regimes: supermartingale, submartingale, and martingale regimes. These three regimes represent the cases where the packets's distance-to-destination tends to decrease, increase, and stay constant, which are similar to the gambling game scenarios in which the expected gambler's asset decreases, increases, and remains constant over time, respectively. Thus, we develop a gambling game-based model to theoretically analyze the bounds and scaling order of access delay for all the three aforementioned packet mobility regimes. Furthermore, to take into account IoT-based applications with strict delay requirement, we introduce and derive the bounds of successful access rate to quantify the likelihood that a packet can be delivered within a pre-specified access delay constraint. Finally, we design a packet shedding scheme to deliver packets within the access delay constraint while also conserving the total transmission power. Our contributions can be summarized as follows.

- We propose a packet mobility model to capture the packet's delivery progress towards the destination cloudlet.
- We analyze the upper and lower bounds of the access delay and successful access rate metrics to show that supermartingale packet mobility achieves the best delay performance.
- We introduce a packet shedding scheme that fulfills an access delay constraint while minimizing energy consumption, which is suitable for time-critical, low-power IoT devices.

1.2.2 Fast Rendezvous for Spectrum-Agile IoT Devices with Limited Channel Hopping Capability

To facilitate pairwise rendezvous between DSA nodes via channel hopping (CH), we consider the nodes' channel switching time and spectrum sensing capacity limits and capture them into the so-called *spectrum slice graphs*. For each DSA node, a spectrum slice graph describes the set of available channels for the next hop, given the currently-visited channel. Then, the channel hopping is viewed as a random walk over the spectrum graphs. To achieve a fast rendezvous, we define a maximum expected time-to-rendezvous (MTTR) metric and formulate an MTTR minimization problem to assign the optimal transition probabilities of the walks corresponding to the CH process. The transition probabilities are assigned for two different network cases, where DSA nodes have and do not have distinct pre-determined roles, which result in role-based (R-RCH) and symmetric random CH (S-RCH) schemes, respectively. We show how to leverage partial-band Nyquist sensing (PBNS) to enable rendezvous not only on the same channel but also on adjacent ones, to achieve an even faster rendezvous. Finally, we evaluate the rendezvous delay of the PBNS-assisted R-RCH and S-RCH schemes using real-world indoor mobile network traces [130] and show that they can achieve satisfactory rendezvous performance. The contributions are summarized as follows.

- We propose spectrum slice graphs to take into account how the limits on frequency switching time and spectrum sensing capacity may affect the CH process.
- We designed PBNS-assisted R-RCH and S-RCH schemes that achieve fast rendezvous while taking into account the CH capability limits described by the spectrum slice graphs.
- We show that the proposed schemes achieve satisfactory rendezvous success rate even under a mobile environment.

1.2.3 Impacts of Dropout-Induced Contractions in D2D-Based Mobile Social Networks

To analyze the impacts of node dropouts in D2D-based MSN, we start by proposing a probabilistic model to highlight the difference between communication and social-based dropouts as the root causes of the network contraction process. Specifically, we introduce (α, β, c) and $(\alpha, \beta, \bar{c})$-systems to model D2D-based MSNs with

1. Social-to-communication ratio α,
2. Nodes to be dropped out with likelihood proportional to the number of their neighbors raised to an exponent β, and
3. Social or communication-based dropouts with symbols c and $\bar{c}$, respectively.

Then, we employ percolation theory [38]—which describes the behavior of connected clusters in a graph after the removal of edges and nodes—to model the temporal progress of the contraction process. To quantify the network-wide impact of initial dropouts, we define a *minimum utility ratio* metric, which is the portion of functional nodes at the end of a contraction process, and then analyze its scaling order with respect to various network parameters. Furthermore, as a user-centric QoE measure, we propose a *node resilience* to quantify the likelihood that a user can enjoy a stable network connection during its ongoing social networking session. We derive the bounds of node resilience as well as its scaling order. These analytical results are verified numerically using real-world [15] and synthetic D2D-based MSN traces, which further highlight their usefulness towards the design of countermeasure schemes for alleviating the extent of a dropout-induced contraction process. Our contributions are outlined as follows.

- By using user activity traces of Facebook users, we found that highly-connected social nodes are dropped out more often (i.e., has an exponent $\beta < 0$) in $(\alpha, \beta, \bar{c})$-system. Otherwise, node dropouts can be random or targeted towards highly-connected communication nodes (i.e., $\beta \geq 0$) in (α, β, c)-system.
- Based on the scaling order analysis, we propose a `MinUtil` that enhances the minimum utility ratio by rejecting and admitting new users when the initial dropouts are severe and mild, respectively.
- We show that the node resilience metric can be increased by suppressing the number of initial dropouts and adding more nodes to the D2D-based MSN.

1.2.4 Characterization of Edge-Induced Cascade-of-Failures in Cyber-Physical Networks

To study the impact of edge breakdown-induced cascade-of-failures in interdependent cyber-physical networks, we present an analytical framework to model how edge fault-induced cascade-

of-failures progresses. In particular, we consider two types of faults: when physical edges are disconnected (i.e., `Type-1`) and when both edges and cyber nodes break down (i.e., `Type-2`). Initial faults, which occur at time $t = 0$, cause nodes in the physical domain to fail. We model the set of non-failure physical nodes that remain functional as a time-decreasing process $\{\mathcal{P}_t : t \geq 0\}$. Each physical node excluded from $\{\mathcal{P}_t : t \geq 0\}$ triggers another process in the cyber domain that will eventually be acted back upon the physical domain, thus causing a cascade-of-failures. We characterize the influence of the cascade by measuring the fraction of physical nodes that remain functional eventually after the cascade-of-failures, which is referred as *node yield*. Then, we use mathematical analysis and numerical evaluation to understand how the node yield is affected by the cyber-physical networks' structure and the amount of edges initially at fault. Further, we define and analyze the *critical disconnection ratio*, which quantifies the smallest fraction of edge disconnections that leads to a complete failure and measures the cyber-physical networks' robustness against disconnections. Finally, we analyze the scaling order of *cascade length* that quantifies the amount of available time until a cascade-of-failures reaches all the failed nodes. Our contributions are three-fold.

- We find that the node yield under `Type-1` fault is higher than than under `Type-2` fault, which indicates guarding against edge disconnections is less important than protecting nodes against initial breakdowns.
- We show that a higher critical disconnection ratio can be achieved by increasing the number of physical edges and guarding cyber nodes against initial breakdowns.
- We found that, unlike the monotonically-decreasing node yield, cascade length is an increasing-then-decreasing function of the amount of disconnected edges. The maximum cascade length is achieved at the critical disconnection ratio.

1.3 Outline and Organization

The rest of this dissertation is organized as follows. Chapter 2 presents how to quickly and efficiently access a cloudlet in IoT-based applications. Chapter 3 presents to achieve fast rendezvous between DSA nodes with limited channel hopping capability. Chapter 4 presents our research results on the modeling and evaluation of how dropout-induced network contractions impact the connectivity of D2D-based MSNs. Chapter 5 investigates the evolution of cascading failures triggered by edge disconnections in cyber-physical networks. Finally, Chapter 6 concludes our research results and discusses the possible future directions.

Chapter 2

Fast and Efficient Access of Cloudlet for IoT-Based Applications in Hostile Environments

2.1 Background and Motivation

The rapid proliferation of smart everyday objects, (i.e., *things*)—such as smart phones, wearables, sensors, and radio frequency identification (RFID) tags—have provided enormous assets for harvesting a massive amount of information around us [7]. These smart objects are equipped with the ability to communicate and form networks with each other—made possible by recent advancements in short-range low-power device-to-device (D2D) technologies [9]—which enables the harvested information to be collected, stored, processed, and presented in a seamless, efficient, and easily-presentable way. Such a new paradigm of collecting, processing, and presenting data, widely-known as the Internet-of-things (IoT), has attracted many new applications in the context of home automation, e-healthcare, smart energy system, and smart cities [78]. It is predicted that among the 28 billion connected devices world-wide in 2021, nearly 16 billion will be IoT objects [8], contributing to $11 trillion of yearly economic impact by 2025 [10].

IoT paradigm hinges upon three main layers [78] (see Fig. 2.1): (i) hardware, made up of sensors, actuators, and embedded communication modules, (ii) middleware, i.e., on-demand storage and computing tools for data analysis, and (iii) presentation, consisting novel visual and interpretation tools. The seamless operation of these three layers relies on the critical first stage, in which sensing data obtained by the "things" must be delivered to and aggregated at the middleware in forms of *cloudlets* [63]. Here, a cloudlet acts as a proxy of a centralized processing algorithm residing on a remote cloud and is installed on a wireless network gateway (i.e., access point) near the "things." Unfortunately, most cloudlet-based IoT studies assume

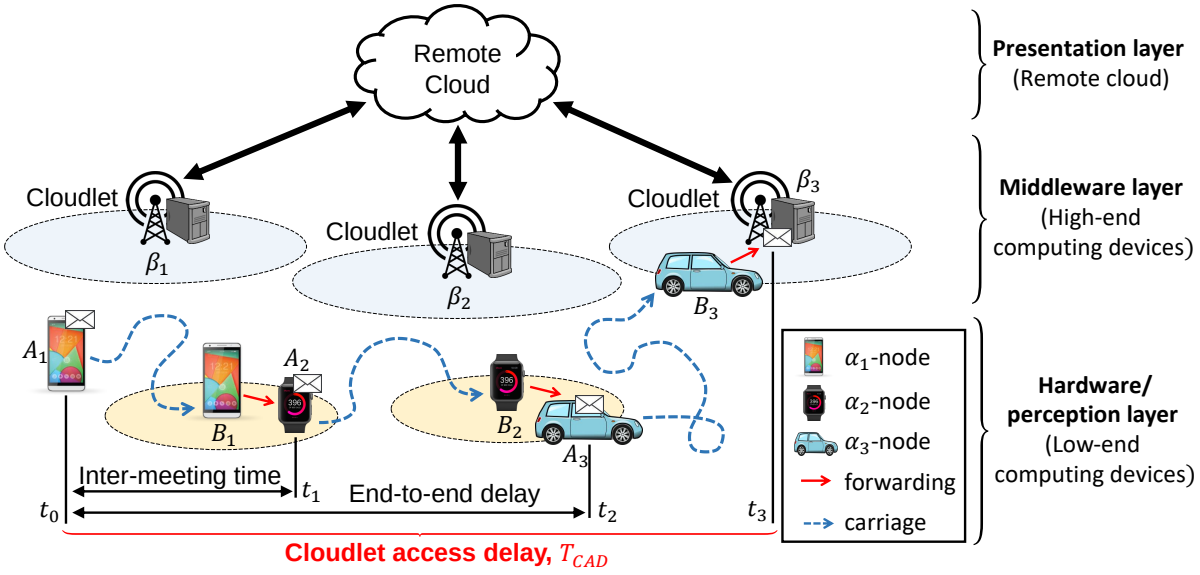


Figure 2.1: Packet delivery in IoT networks under hostile environments.

that the network area is sufficiently covered by the cloudlets and the device-to-cloudlet links (i.e., the first stage) are stable [63], which simply does not hold in challenging environments such as disaster areas.

Consider an area after a 9.1-Richter magnitude earthquake. To build an up-to-date post-disaster map, first responders turn on GigaPan sensing [66], which takes as input the close-up pictures of disaster scenes uploaded by local residents using off-the-shelf mobile phones. These mobile phones, whose cellular and WiFi connections had gone out-of-service due to infrastructure damage, can send the pictures to neighboring devices, which then forward the images to a cloudlet using multi-hop communication. At the cloudlet, the received pictures are stitched together to build high-fidelity panoramic images. As new maps of the disaster area are constructed from the images, rescue efforts speed up and become more effective [157].

In a disaster area, IoT devices are highly power constrained, the cloudlets are sparsely-located, and hilly terrain limits the wireless signal's range [78, 112]. Despite the power, coverage, and shadowing limitations, challenging environments are envisioned to host important applications such as search-and-rescue mission. To support these missions, every "thing" periodically senses their surrounding, encapsulates the sensing data into a packet ϵ, and then sends the packet to its neighbors. These neighboring nodes can further store ϵ to their internal buffers, carry it as they move in the network, forward ϵ to any nearby nodes, and then repeat this *store-carry-forward* (SCF) process [174] until ϵ can be delivered to a cloudlet. As a result, packet delivery for cloudlet-based IoT in challenging environments shares several resemblances with existing techniques that use (i) multi-hop communication, similar to mobile ad-hoc networks

(MANETs) [67], and (ii) SCF, as in delay-tolerant networks (DTNs) [91].

There have been extensive studies that examine the packet delivery performance of MANET and DTN in terms of delay [174, 76, 119], which by large can be explained by the generic model depicted in Fig. 2.1. For instance, let a mobile node α_1 generates a unicast packet ϵ at time t_0, with a target destination client of α_3. Node α_1 then carries ϵ while being mobile and, at time t_1, forwards the packet to a nearby α_2-node at location A_2. This one-hop delay from t_0 to t_1 is called the *inter-meeting time* [119, 76, 174]. To proceed, packet ϵ is stored at and carried by α_2 until it can be sent to the target destination α_3 located at A_3. The time from when ϵ is generated until it is delivered to α_3, i.e., $(t_2 - t_0)$, is called the *end-to-end delay* and has been considered as one of the main quality-of-service metrics in MANET and DTN studies [174].

Unlike MANET/DTN with end-to-end delay metric described above, cloudlet-based IoT in challenging environments have several fundamental differences. Firstly, the data collected by "things" (α-nodes) must be uploaded to a cloudlet for pre-processing. As a result, the SCF process must continue until packet ϵ can reach a cloudlet, i.e., the α_3-node in Fig. 2.1 can eventually deliver ϵ to β_3 via carry-and-forwarding. Thus, we are more concerned about the *cloudlet access delay* (CAD), i.e., $(t_3 - t_0)$.[1] Secondly, there are multiple cloudlets in the network and packet ϵ is considered to be delivered if it can reach any of the cloudlets, because each cloudlet is a proxy of the central Cloud and can locally process the sensed data. Instead of associating packet ϵ generated at A_1 with the nearest β_1-node (see Fig. 2.1), the packet ϵ can reach another β_3-node faster and should be associated with this cloudlet. Thirdly, the applications supported by cloudlet-based IoT, such as public services for search-and-rescue [66, 78], are time-critical and non-delay-tolerant. In these applications, the information embedded in packet ϵ must reach the cloudlet within a certain delay bound and otherwise will be useless. Due to the aforementioned fundamental differences, existing studies regarding the end-to-end packet delivery delay with pre-determined source and destination nodes for delay-tolerant networks [174, 76, 119] cannot be applied directly to our cloudlet-based IoT in hostile environments.

In addition to supporting time-critical applications, IoT networks need to conserve power consumption thanks to the capacity limits of energy storage device carried by the "things" [78]. Achieving the former time-critical goal, which is equivalent to minimizing delay, can be achieved via a more complicated routing scheme (i.e., by increasing path redundancy via multiple packet copies [91]), but contradicts the latter power conservation goal. Thus, two important questions remain wide-open: *"What is the CAD of the cloudlet-based IoT in challenging environment, and how to support time-critical applications while minimizing power consumption?"*

To address the questions above, we start by reviewing the existing literature in Section 2.2.

[1] Notice that the end-to-end delay is much larger than the cloudlet-to-cloud delay via wired internet or proprietary network [91], while the mobile-to-cloudlet delay (CAD) has the same order as the end-to-end delay. Thus, instead of focusing on the overall mobile-to-cloudlet-to-cloud delay, we are more interested on analyzing the CAD.

Then, we propose a *packet mobility model* to capture the trajectory of packet ϵ's movement over time in Section 2.3. In Section 2.4, we outline the similarities of the packet mobility to a gambling game, and accordingly show that the packet mobility for cloudlet-based IoT can be divided into three cases: each exhibiting either supermartingale, martingale, or submartingale-like properties. In Section 2.5 we mathematically derive the bounds as well as the scaling order of the expected CAD for the three packet mobility cases. Based on the theoretical packet mobility formulation, we design a packet shedding scheme for minimizing the average per-message power consumption in Section 2.6. Finally, we conclude in Section 3.6 and defer all major proofs to the Appendices.

2.2 Related Works

After outlining the motivation behind studying CAD, we state the key differences between our access delay study to the existing results in the literature.

Delay has been considered as a major performance measure for wireless systems, especially intermittently-connected mobile networks [174]. As an initial effort towards analyzing delay, early studies have examined the *inter-meeting time* [76, 77]. To this end, many researchers simply assume exponentially-distributed inter-meeting times [77], which is corroborated by numerical simulations [76] based on random waypoint (RWP) node mobility [42]. However, newer results via real mobility traces indicate that the tail behavior of the inter-meeting time is close to a power law [44], instead of exponential.

As suggested by Fig. 2.1, the inter-meeting time is equivalent to the *one-hop delay*, that is, the time until a packet sent by a source node can be delivered to a destination node when multi-hop routing is employed. As a result, the inter-meeting time has been widely used as a building block for analyzing the end-to-end delay in fully-distributed DTNs without infrastructure support. For example, Markov chains [76] and ordinary differential equations [174] have been applied to analyze the end-to-end delay with respect to an independently- and identically-distributed (i.i.d.) inter-meeting time for networks employing epidemic routing. Moreover, under a more generalized heterogeneous inter-meeting time, the delay of broadcasting a packet to all the nodes in MANETs employing epidemic routing has also been analyzed using Markovian models [119].

In view of end-to-end delay, however, the endpoints of a packet are mobile nodes [140, 120]. Hence, end-to-end delay is more suitable for evaluating *peer-to-peer* (mobile-to-mobile) applications such as file transfer and instant messaging [2, 121]. A more modern and common paradigm of wireless applications, however, follows a *client-server* paradigm where mobile nodes as clients connect to a server-hosted service to access media-rich resources [98]. For example, collecting data from sensors in the ocean [136], other than building a post-disaster map by stitching pic-

tures snapped by respondents [66], are some of the services with a client-server paradigm. Since these services are typically hosted on servers interconnected to the infrastructure nodes through low-latency interconnections, we are more interested in studying the *mobile-to-infrastructure* delay, instead of the end-to-end delay.

In this chapter, we analyze the mobile-to-infrastructure delay of cloudlet-based IoT applications in challenging environments, which we refer as the cloudlet access delay (CAD). Our analysis in Section 2.5 takes into account the initial packet-to-cloudlet distance for deriving the bounds and scaling order of CAD. Thus, our results are more generic than existing access delay analysis [83], in the sense that the existing analysis can also be obtained by averaging our CAD results over all the possible initial packet locations (i.e., distances).

In time-critical IoT applications, the mobile α-nodes rely on their battery and/or energy-harvesting unit and are highly power-constrained [78]. Although many energy-efficient packet delivery schemes for wireless sensor networks have been proposed (see [113] and the references therein), they all assume a connected network and are not suitable for IoT networks with intermittent connectivity. Motivated by the lack of delivery schemes for challenging environments, in Section 2.6.2 we propose a power-efficient prediction-based scheme that discards the packets which cannot be delivered on-time. Our objective is similar to that of an energy-efficient probabilistic routing proposed in [101].

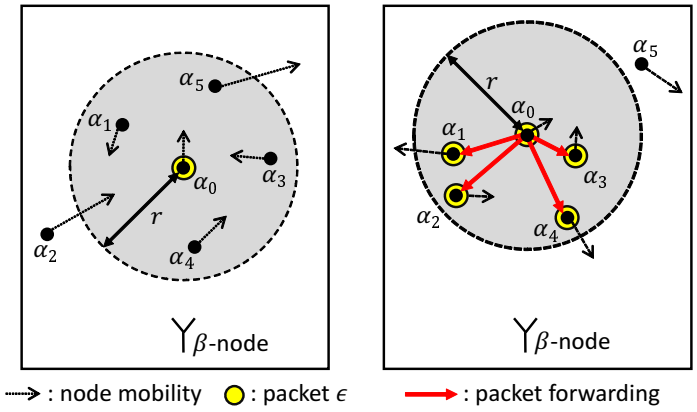
The existing probabilistic scheme, however, considers either two-hop or epidemic routings—which either induces long delay or large overhead, while the proposed *packet shedding* scheme uses geographic-based heuristic routing [21, 99] that is less complex than epidemic routing while also having a delay performance better than two-hop routing. Moreover, the proposed packet shedding can be extended to accommodate generic routing schemes.

2.3 Models and Problem Formulation

In this section, we outline the network model under hostile environments, and then present a *packet mobility model* to capture the movements of data packets from IoT devices to the cloudlets. Then, we state our main objectives.

2.3.1 Network and Communication Models

We consider a set $\mathcal{A} = \{\alpha_0, \ldots, \alpha_{n-1}\}$ of n mobile α-nodes representing the "things" (i.e., smart IoT devices) located within a rectangular network area of $\mathcal{N} = [0, L]^2$. Each α-node is aware of its own location, either via GPS reading or low-power inertial navigation system [99]. Moreover, the IoT devices (α-nodes) are not connected to stable power sources and need to conserve their energy usage by limiting transmit power. Accordingly, two α-nodes can communicate and are



(a) At time $t = 0$. (b) At time $t = 1$.

Figure 2.2: Packet carriage and forwarding in a D2D-Cloudlet.

said to be *in contact* if they are within a small transmission range r of each other.

In an IoT paradigm, the α-nodes periodically harvest data from its surroundings (e.g., in form of crowd-sourced pictures in GigaPan [66]). The collected data are encapsulated as a packet ϵ and then sent to wireless edge gateways that are connected to a remote cloud via bakchaul in forms of proprietary links or internet connectivity. Unfortunately, the backhaul can be overloaded, i.e., when many users try to access Internet through the gateway simultaneously, and even become fragile due to infrastructure collapse [129]. In this hostile network setting, the concept of *mobile edge computing* [115] that leverages *cloudlets*—proxies of the real cloud that are capable of pre-processing the collected data locally—can be used to alleviate the backhaul fragility and overload [128]. The cloudlets are installed as a middleware on portable edge computing platforms [63], denoted here as β-nodes, which also act as wireless gateways that replace the damaged WiFi access points and LTE base stations. We assume these β-nodes are uniformly-located, dividing the wireless network area $\mathcal{N}$ into $l \times l$ cells, where $l \leq L$.

We further assume that the β-nodes adopt the same transmission range r as the α-nodes, and r is much smaller than the cells' side length l. As a result, an end-to-end connected path to deliver packet ϵ to any of the β-node may not exist at any time. However, as the α-nodes are moving, they can come into contact with other nodes that eventually reach a β-node. To leverage this mobility-induced opportunity to incrementally deliver packet ϵ, we employ the following approach. Let the network lifetime be represented as a collection of discrete non-negative time slots $\mathcal{T}$ and $\mathbf{1}_{\alpha_k}(t)$ indicates whether α_k carries packet ϵ at time $t \in \mathcal{T}$. Consider an event-driven process, as exemplified in Fig. 2.2a, where ϵ is generated by and stored in α_0 at time $t = 0$. Initially, $\mathbf{1}_{\alpha_0}(0) = 1$ while the other entries are equal to zeroes. In the next time slot, $t = 1$, the packet is *carried* by the mobile node into a new location, which is indicated as the

center point of node α_0 in Fig. 2.2b. At this time slot (and also the subsequent ones), the set of packet-carrying nodes $\{\alpha_k : \mathbf{1}_{\alpha_k}(1) = 1\}$ does not assume any particular packet routing scheme. However, when packet ϵ is routed in an epidemic fashion [145], it becomes *forwarded* to all the contacts of α_0, which results in $\mathbf{1}_{\alpha_k}(1) = 1$ for $k \in \{0, 1, \ldots, 4\}$. Such a *store-carry-forward* (SCF) process [174] is repeated until packet ϵ is located within the transmission range r of a β-node.

2.3.2 Definition of Packet Mobility

To deliver packet ϵ to a β-node, the most challenging issue is that the packet can travel through multiple paths with different directions and speed, since both the node mobility and forwarding schemes jointly contribute to the journey of a packet. While algorithms for delivering ϵ can be implemented in various ways [145, 139, 67, 77], how to characterize the travel distance, speed, and corresponding delivery time of a packet still does not have an obvious answer. Therefore, we propose a *packet mobility* model, which results from client mobility as well as carry-and-forwarding, as follows.

Definition 1. *Let* $\vec{\mathbf{X}}_\epsilon(t) := \{\vec{X}_{\alpha_k}(t) : \mathbf{1}_{\alpha_k}(t) = 1\}$ *be the locations of the packet-carrying* α*-nodes at time* $t \in \mathcal{T}$*. Then, packet mobility is defined as a collection* $\{\vec{\mathbf{X}}_\epsilon(t) : t \in \mathcal{T}\}$*.*

Let $\vec{X}_\beta \in R^2$ be the location of the β-node closest to the packet, and denote a *distance-to-destination* metric as

$$L_\epsilon(t) := \min_{\vec{X}_{\alpha_k}(t) \in \vec{\mathbf{X}}_\epsilon(t)} \left\| \vec{X}_{\alpha_k}(t) - \vec{X}_\beta \right\|, \tag{2.1}$$

where $\|\cdot\|$ denotes the Euclidean distance. Then, by comparing the distance-to-destination between adjacent time steps, we obtain a simplified, one-dimensional metric for tracking the packet delivery progress as follows.

Definition 2. *Packet propagation speed at time* $t \in \mathcal{T}$ *is defined as the distance-to-destination's increment between two subsequent time slots, that is,*

$$v_t := L_\epsilon(t) - L_\epsilon(t-1). \tag{2.2}$$

The relation between packet mobility $\{\vec{\mathbf{X}}_\epsilon(t)\}$, distance $L_\epsilon(t)$, and packet propagation speed v_t on a network employing single-copy routing[2] [77] is illustrated in Fig. 2.3.

In Fig. 2.3, due to the mobile edge computing paradigm (see Section 2.3.1), the network is divided into $l \times l$ cell areas, each of them covered by a cloudlet β-node located at the center of the cell. In some circumstances, packet ϵ may be (i) carried by an α-node across the cell

[2]We say that a routing scheme is single copy if $|\vec{\mathbf{X}}_\epsilon(t)| = 1$ for all $t \in \mathcal{T}$. A routing is multi-copy if $\forall t \in \mathcal{T}, |\vec{\mathbf{X}}_\epsilon(t)| > 1$.

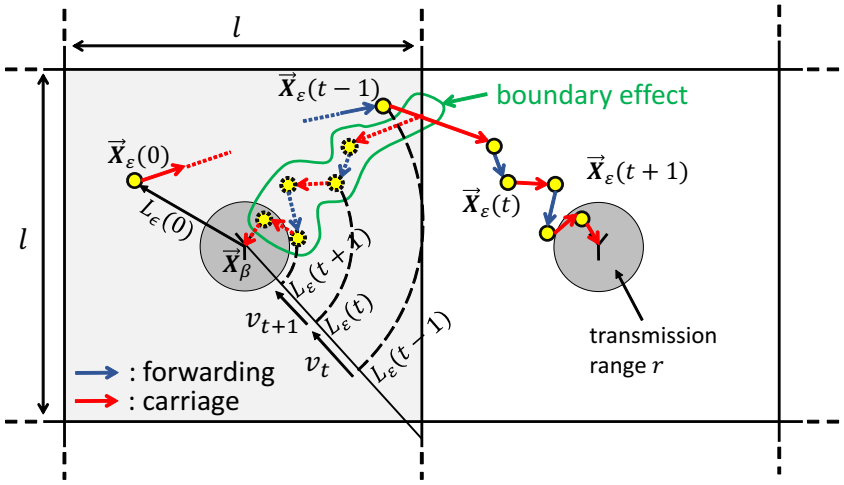


Figure 2.3: Packet mobility $\{\vec{\mathbf{X}}_\epsilon(t)\}$, distance-to-destination $L_\epsilon(t)$, propagation speed v_t, and boundary effect.

boundary or (ii) forwarded to another α-node in an adjacent cell and eventually be delivered to a β-node in this cell. Notice that every β-node is a cloudlet that can process ϵ locally [128], and the processing result from the adjacent β-node can be shared with the original β-node on the left (and vice versa) via the remote cloud. Hence, the exact recipient β-node is not important, and the packet is considered to be delivered upon reaching *any* of the β-nodes. Moreover, the packet's trajectory across the boundary can be mapped back into its original cell, which is taken into account by *boundary reflection mapping* (see the dotted arrows in Fig. 2.3). In this chapter, we refer the impacts of boundary reflection to the packet mobility and its metrics (i.e., $L_\epsilon(t)$ and v_t) as a *boundary effect*. We will show in Section 2.5.2 that boundary effect can help accelerate packet delivery to a cloudlet.

2.3.3 Problem Formulation

In cloudlet-based IoT applications under challenging environments, i.e., disaster areas, the sensed data generated by smart devices in forms of a packet ϵ must be delivered to the cloudlet β-node in a timely manner. The time to deliver packet ϵ is rigorously defined as follows.

Definition 3. *Cloudlet access delay (CAD) is the time duration from the moment a packet ϵ is generated at time $t = 0$ until it reaches at least one β-node, denoted as*

$$T_{CAD} := \inf_{t \in \mathcal{T}} \left\{ t : L_\epsilon(0) + \sum\nolimits_{k=1}^{t} v_k < r \right\}. \tag{2.3}$$

Unlike the conventional end-to-end delay [174], T_{CAD} measures the mobile-to-cloudlet delay in IoT-based applications (see Fig. 2.1). We are not concerned about the inverse direction since

in IoT the information processed at cloudlets are presented by the cloud to other entities [78], not the "things" (mobile devices).

To support emergency IoT applications in disaster areas, the CAD must satisfy a certain *delay requirement* of $\tau \geq 1$ time steps. For example, first responders need to reach an emergency location within 5 minutes to save a victim's life after he/she requested 911 assistance [88], which can be integrated into future IoT-based services. In this case, the user-generated emergency request packet must be delivered in much less than $\tau = 300$, for time steps with one second resolution, in order to rescue the victim's life.

In addition to strict delay requirement, cloudlet-based IoT applications in challenging environments have stringent power constraint [78] that is mainly limited by the capacity of batteries powering the mobile α-devices. Since α-devices' access to stable electricity source is highly restricted, power becomes a vital resource that should be managed carefully.

In terms of the α devices' power consumption, a significant amount may be attributed to wireless transmissions for delivering packets [80]. Suppose a packet ϵ is generated at time $t = 0$. Depending on the employed routing scheme, there can be multiple copies of ϵ and each copy may or may not be forwarded to another node at any particular time. Let $P_f(t, j) > 0$ be the transmit power consumed when the j^{th} copy of ϵ is forwarded at time $t \in \mathcal{T}$ from a packet-carrying α_{k_1}-node to (i) another α_{k_2}-node or (ii) a β-node. Otherwise, $P_f(t, j) = 0$ if ϵ is not forwarded but carried. Recall that there are $|\vec{\mathbf{X}}_\epsilon(t)|$ copies of ϵ at time t. Let packet ϵ be delivered at time T_{CAD}. We assume that once a copy of ϵ is delivered, the β-node sends a broadcast message so that any packet-carrying α-node drops ϵ immediately. Then, the *per-message transmit power consumption* can be given as

$$P_\epsilon = \sum_{t=1}^{T_{CAD}} \sum_{j=1}^{|\vec{\mathbf{X}}_\epsilon(t)|} P_f(t, j). \tag{2.4}$$

With T_{CAD} and P_ϵ at hand, we are ready to state our main research question: *"How large is the CAD (T_{CAD}) for IoT-based networks in challenging environment? How to satisfy the delay constraint τ while also minimizing the per-message power draw P_ϵ?"* We will first characterize the CAD in Sections 2.4-2.5, and proceed to optimize P_ϵ in Section 2.6.

2.4 Packet Mobility Modeling and Analysis

Before analyzing the CAD, we model the packet mobility as a gambling game, which exhibits either supermartingale, martingale, or submartingale-like properties. These properties will be used as a foundation to analyze the bounds and scaling laws of CAD in Section 2.5.

2.4.1 A Gambling Game Perspective on Packet Mobility

In a cloudlet-based IoT application, a packet ϵ is generated at a mobile α-node with an initial distance $L_\epsilon(0)$ from a cloudlet β-node. Due to the SCF procedure, ϵ travels according to a packet mobility $\{\vec{\mathbf{X}}_\epsilon(t) : t \in \mathcal{T}\}$, until it is located within the range r of a cloudlet. Upon taking a closer look at such a packet delivery progress, we observe the following properties.

1. **The process $\{\vec{\mathbf{X}}_\epsilon(t)\}$ has one eventual outcome: the SCF procedure wins.** Specifically, the SCF can bring ϵ close enough to the β node, in that the distance-to-destination $L_\epsilon(t)$ becomes *less* than a threshold r.

2. In order to win, **the SCF procedure needs to cumulatively gather reward in form of $L_\epsilon(t)$.** Whenever the SCF is in effect, a certain amount of packet propagation speed v_t contributes to the overall distance, that is, $L_\epsilon(t) = L_\epsilon(0) + \sum_{k=1}^{t} v_k$.

By considering the SCF as an anti-gambler (similar to a dealer in Blackjack), an imaginary entity as a gambler, and $L_\epsilon(t)$ as as the gambler's asset, the problem of delivering ϵ from IoT devices to a cloudlet becomes similar to a gambling game.

Accordingly, we construct a game of gamblers as follows. Let the gambler starts with an asset of $L_\epsilon(0)$, and refer the packet mobility due to the SCF process during one time slot as a *play*. Let $L_\epsilon(t)$ be the gambler's asset, which is increased by v_t during the t^{th} play. In order for the SCF to win and *ruin* the gambler's fortune, both entities must participate in each play until $L_\epsilon(t)$ is less than the bankruptcy threshold r.

It has been shown that martingale, supermartingale, and submartingale theories [90]—branches of modern probabilistic measure theory—are effective tools for solving the gambler's ruin problem. In order to leverage these theories, the gambler's asset, $\{L_\epsilon(t) : t \geq 1\}$, must satisfy the following properties.

Definition 4 (Martingale)**.** *Let a sequence of σ-algebras $\{\mathcal{F}_t\}$ be called a filtration if $\mathcal{F}_t \subset \mathcal{F}_{t-1}$ for any $t \in \mathcal{T}$. Then, a process $\{L_\epsilon(t)\}$ is called a martingale relative to a filtration $\{\mathcal{F}_t\}$ if (i) $L_\epsilon(t)$ is $\mathcal{F}_t$-measurable, (ii) $E|L_\epsilon(t)| < \infty$ for any $t \in \mathcal{T}$, and (iii) $E(L_\epsilon(t)|\mathcal{F}_{t-1}) = L_\epsilon(t-1)$ almost surely.*

Definition 5 (Supermartingale and submartingale)**.** *A process $\{L_\epsilon(t)\}$ is a supermartingale relative to a filtration $\{\mathcal{F}_t\}$ if (i) $L_\epsilon(t)$ is $\mathcal{F}_t$-measurable, (ii) $\forall t \in \mathcal{T}, E|L_\epsilon(t)| < \infty$, and (iii) $E(L_\epsilon(t)|\mathcal{F}_{t-1}) < L_\epsilon(t-1)$ almost surely. Otherwise, if (i) and (ii) hold and $E(L_\epsilon(t)|\mathcal{F}_{t-1}) > L_\epsilon(t-1)$ almost surely, then $\{L_\epsilon(t)\}$ is a submartingale.*

In the following, we show that the gambler's asset, which corresponds to the packet mobility, exhibits either supermartingale, martingale, or submartingale-like properties.

2.4.2 Analysis of Packet Mobility Properties

According to Definitions 4 and 5, examining the supermartingale, martingale, and submartingale properties boils down to checking three conditions as follows. The first two conditions are easy to check.

[C1] **$L_\epsilon(t)$ is $\mathcal{F}_t$-measurable:** Let $(\Omega, \mathcal{F}, P)$ be a probability space in that Ω is a set, $\mathcal{F}$ is a σ-algebra, and P is a probability measure on $(\Omega, \mathcal{F})$ [90]. Consider a natural filtration $\mathcal{F}_t := \sigma(L_\epsilon(0), \ldots, L_\epsilon(t))$, where $\sigma(\mathcal{L})$ is the smallest σ-algebra containing set $\mathcal{L} \subset \Omega$. Then, it is obvious from our construction that $\{L_\epsilon(t)\}$ is relative to $\mathcal{F}_{t-1}$ and $L_\epsilon(t)$ is $\mathcal{F}_t$-measurable.

[C2] **$E|L_\epsilon(t)| < \infty$ for all $t \in \mathcal{T}$:** We have $E|L_\epsilon(t)| = E|L_\epsilon(0) + \sum_{k=1}^{t} v_k| \leq E|L_\epsilon(0)| + tE|v_k|$. Since $L_\epsilon(0)$ is finite, it suffices to show $E|v_k| < \infty$. Observe that the maximum distance traversed by packet ϵ at any time slot is equal to the transmission range r plus the node velocity per time slot v_{node}. Since r and v_{node} are both finite, $E|v_k| \leq r + v_{node} < \infty$.

On the other hand, verifying the third condition, as outlined below, is non-trivial.

[C3] **$E(L_\epsilon(t)|\mathcal{F}_{t-1})$ versus $L_\epsilon(t-1)$:** In supermartingale, martingale, and submartingale cases, $E(L_\epsilon(t)|\mathcal{F}_{t-1})$ must be smaller than, equal to, and larger than $L_\epsilon(t-1)$, respectively. These criteria are non-trivial to verify due to the two following challenges. Firstly, for generic packet mobility—consisting an arbitrary combination of node mobility and packet routing schemes—the closed-form relation between the current distance-to-destination $L_\epsilon(t)$ to the previous one, $L_\epsilon(t-1)$, may not be readily available. Secondly, the relation may be non-homogeneous and varies depending on the exact value of $L_\epsilon(t-1)$.

To alleviate the two aforementioned challenges, we present mathematical and numerical-based methods to check the third condition (C3) as follows.

Mathematical model for checking C3

Mathematical analysis can be used when the closed-form relation between $L_\epsilon(t)$ and $L_\epsilon(t-1)$ is available. One example is when *random walk* (RW) node mobility [42], simply referred as T1, and *no packet routing* (R1) is employed. In RW mobility (T1), each α-node selects to move to a new location at every time slot, by randomly choosing a velocity and a direction from uniform distributions, respectively. The following result holds when T1 is combined with R1.

Let packet ϵ be located at a point with a distance $L_\epsilon(t-1)$ from the β-node. We can compute the probability density function (PDF) of ϵ's location at the next time step, $L_\epsilon(t)$, by dividing the number of points corresponding to each $L_\epsilon(t)$ with the union of all possible future locations. The PDF can then be weighted by $L_\epsilon(t)$ to obtain the expected value $E(L_\epsilon(t)|L_\epsilon(t-1))$. Finally,

we can verify that the submartingale property holds by showing that $E(L_\epsilon(t)|L_\epsilon(t-1)) > L_\epsilon(t-1)$ for all $L_\epsilon(t-1)$. The result is given below.

Lemma 1. *A network with smart devices moving according to RW mobility (`T1`) and employing direct contact routing (`R1`) has a submartingale packet mobility.*

Proof. We want to show $E(L_\epsilon(t)|\mathcal{F}_{t-1}) > L_\epsilon(t-1)$ geometrically, which is motivated by a similar approach to analyze the end-to-end delay in mobile ad-hoc networks [83]. Unlike the existing approach that considers the packet's distance to sender, however, we focus on the distance $L_\epsilon(t)$ between the packet to the destination β-node. Moreover, we are interested in calculating the expected packet propagation progress $E(L_\epsilon(t)|\mathcal{F}_{t-1}) > L_\epsilon(t-1)$, instead of the end-to-end delay [83], as detailed below.

Since direct contact routing is used, the packet mobility is mainly due to the random walk node mobility. In such a mobility model, a message-carrying α_i-node moves from its previous location $\vec{\mathbf{X}}_\epsilon(t-1) = \{\vec{X}_{\alpha_i}(t-1)\}$ to a new location by randomly choosing a speed v_{node} (*meters/timeslot*) and a direction θ (*radians*) uniformly at random from ranges $[v_{min}, v_{max}]$ and $[0, 2\pi]$, respectively. Thus, the location of node α_i at the next time slot, $\vec{\mathbf{X}}_\epsilon(t) = \{\vec{X}_{\alpha_i}(t)\}$, is within a ring area $\{c_2^\alpha - c_1^\alpha\}$ where c_2^α and c_1^α are the arcs centered at $\vec{\mathbf{X}}_\epsilon(t-1)$ with radii $r_2 = v_{max}$ and $r_1 = v_{min}$, respectively, as depicted in Fig. 2.4. Furthermore, let $\Pr\{y_\epsilon|x_\epsilon\} := \Pr\{L_\epsilon(t) = y_\epsilon|L_\epsilon(t-1) = x_\epsilon\}$ be the *progress probability* that node α_i located at $\vec{\mathbf{X}}_\epsilon(t-1)$ with a distance-to-destination x_ϵ moves to a new location $\vec{\mathbf{X}}_\epsilon(t)$ with a distance-to-destination of y_ϵ. Then,

$$E(L_\epsilon(t)|\mathcal{F}_{t-1}) = E(L_\epsilon(t)|L_\epsilon(t-1) = x_\epsilon) = \int_{x_\epsilon - r_2}^{x_\epsilon + r_2} y_\epsilon \Pr\{y_\epsilon|x_\epsilon\} dy_\epsilon. \tag{2.5}$$

As a result, comparing $E(L_\epsilon(t)|\mathcal{F}_{t-1})$ to $L_\epsilon(t-1)$ becomes the problem of calculating the progress probability $\Pr\{y_\epsilon|x_\epsilon\}$.

To calculate the progress probability, let c_2^β be an arc with radius y_ϵ centered at the β-node's location $\vec{\mathbf{X}}_\beta$. Let $\gamma(x_\epsilon, y_\epsilon)$ be the length of the intersection between the arc c_2^β with the ring area $\{c_2^\alpha - c_1^\alpha\}$, as shown in Fig. 2.4. Then, the progress probability can be re-stated as $\Pr\{y_\epsilon|x_\epsilon\} = \frac{\varphi x_\epsilon}{\pi(r_2^2 - r_1^2)}$, where $\varphi = \varphi_2 - \varphi_1, if$—$\mathrm{y}_\epsilon - x_\epsilon| \leq r_1$,
$\varphi_2, if \mathrm{r}_1 < |y_\epsilon - x_\epsilon| \leq r_2$; φ_2 is the angle formed by the the point $\vec{\mathbf{X}}_\epsilon(t-1)$, the center of the cell $\vec{\mathbf{X}}_\beta$, and the outermost point of arc $\gamma(x_\epsilon, y_\epsilon)$; and φ_1 is the angle formed by the two former points and the *innermost* point of arc $\gamma(x_\epsilon, y_\epsilon)$ after removing the area $c(\vec{\mathbf{X}}_\epsilon(t-1), r_1)$. Here, "innermost" refers to a point of an arc closest to the line connecting the center point $\vec{\mathbf{X}}_\beta$ and $\vec{\mathbf{X}}_\epsilon(t-1)$, while $c(\vec{\mathbf{X}}, r)$ denotes a solid circle centered at $\vec{\mathbf{X}}$ with radius r.

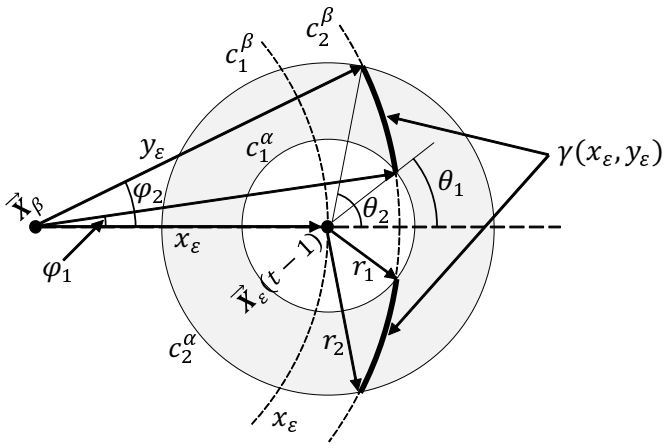


Figure 2.4: Geometry of a D2D-Cloudlet employing random walk node mobility and direct contact routing.

Next, we define θ_1 and θ_2 as two new angles as depicted in Fig. 2.4 to obtain

$$(x_\epsilon + r_1 \times \cos(\theta_1))^2 + (r_1 \times \sin(\theta_1))^2 = y_\epsilon^2, \tag{2.6}$$

$$x_\epsilon + r_1 \times \cos(\theta_1) = y_\epsilon \times \cos(\varphi_1). \tag{2.7}$$

Substituting (2.7) into (2.6) results into a closed-form solution $\varphi_1 = \arccos\left\{\frac{x_\epsilon^2+y_\epsilon^2-r_1^2}{2x_\epsilon y_\epsilon}\right\}$, while a similar solution for φ_2 can be acquired by substituting $(\cdot)_1$ with another subscript $(\cdot)_2$. Finally, we obtain $E(L_\epsilon(t)|\mathcal{F}_{t-1}) > x_\epsilon$ after plugging φ_1 and φ_2 into (2.4.2) and into the progress probability $\Pr\{y_\epsilon|x_\epsilon\} = \frac{\varphi x_\epsilon}{\pi(r_2^2-r_1^2)}$, and then solving the integral in (2.5) numerically. Such an inequality indicates that the resulting packet mobility is submartingale-like. □

Mathematical analysis can also be used for the Infostation model [67]. In this model, α-nodes are represented as emergency vehicles distributed over a straight road segment. If the Infostation model is employed in conjunction with *randomized routing* (R2) [139], which is a single-copy routing where the packet-carrying α-node forwards the packet to a randomly-selected neighbor until a β-node is encountered, we obtain the following corollary.

Corollary 1. *Consider a network with α-nodes distributed uniformly with homogeneous velocities on a linear road segment. Let β-nodes be equally-spaced road-side units [67]. If randomized routing (R2) is employed, then the resulting packet mobility is martingale.*

Proof. Recall that the network area is a line segment, the two-way traffic has identical velocities v_{node}, and the packet ϵ is forwarded to a randomly-picked contact. As a result, the progress probability $\Pr\{L_\epsilon(t) = y_\epsilon|L_\epsilon(t-1) = x_\epsilon\}$ is a uniform distribution with range $[x_\epsilon - (r +$

$v_{node}), x_\epsilon + (r + v_{node})]$. Plugging such a progress probability into (2.5) yields $E(L_\epsilon(t)|\mathcal{F}_{t-1}) = x_\epsilon = L_\epsilon(t-1)$. □

Corollary 1 and the aforementioned Lemma 1 exemplify the case when condition C3 is determinable via mathematical modeling. When the mathematical model is hard to derive, C3 can be verified using numerical analysis, as outlined below.

Numerical analysis for checking C3

General packet mobility cases may employ multi-copy scheme such as *epidemic routing* (R3) [145], in which each packet-carrying α-node *copies* packet ϵ to all other nodes it encounters. As a result, the current distance-to-destination $L_\epsilon(t)$ not only depends on $L_\epsilon(t-1)$ (i.e., the past locations of packet-carrying nodes), but also on the positions of the non-carrier α-nodes. In this case, the task of mathematically analyzing $L_\epsilon(t)$ and comparing it to $L_\epsilon(t-1)$ becomes very challenging. Thus, instead of using mathematical analysis, we evaluate $E(L_\epsilon(t)|\mathcal{F}_{t-1})$ versus $L_\epsilon(t-1)$ via numerical evaluations.

To proceed, we generate a large number of network realizations to calculate the *expected packet propagation speed* $E(v_t|\mathcal{F}_{t-1}) := E(L_\epsilon(t) - L_\epsilon(t-1)|\mathcal{F}_{t-1})$. According to C3, showing that a packet mobility is supermartingale, martingale, and submartingale is equivalent to verifying that $E(v_t|\mathcal{F}_{t-1})$ is smaller than, equal to, or larger than zero, for all possible $L_\epsilon(t-1)$'s. However, this is hard to do in practice since $L_\epsilon(t-1)$ can take any positive real value, with infinite number of possibilities. To relax the infinite domain, we divide the $l \times l$ cell area into rings $\mathcal{R} = \{(a_0, b_0], (a_1, b_1], \ldots\}$, where $a_i \geq r$ and $b_i > a_i$ are the inner and outer diameters of the i^{th} ring (see Fig. 2.5a), which will serve as the "bins" for $L_\epsilon(t-1)$. At each time step, we collect $L_\epsilon(t)$ and $L_\epsilon(t-1)$ to calculate $v_t = L_\epsilon(t) - L_\epsilon(t-1)$. By assuming that for any particular index i the speeds $\{v_t : L_\epsilon(t-1) \in (a_i, b_i]\}$ are independently and identically-distributed (IID), we can leverage the law of large numbers to calculate the *average packet propagation speed* for the i^{th} interval as

$$E(v_t(i)) = \frac{\sum_{t\in\mathcal{T}, L_\epsilon(t)\in(a_i,b_i]} v_t}{\sum_{t\in\mathcal{T}} \mathbf{1}_{(L_\epsilon(t)\in(a_i,b_i])}}, \tag{2.8}$$

where $\mathbf{1}_{(\mathcal{B})} \in \{0, 1\}$ indicates the occurrence of event $\mathcal{B}$. The resulting $E(v_t(i))$ is then averaged over a large number of network realizations. It is easy to verify that as $\gamma := b_i - a_i$ goes to zero, $E(v_t(i))$ converges to $E(v_t|\mathcal{F}_{t-1})$. Based on this asymptotic property, we present the following.

Definition 6. *A packet mobility is supermartingale, martingale, or submartingale-like if* $\forall i, E(v_t(i))$ *is smaller than, equal to, or larger than zero.*

With Definition 6 at hand, we are ready to discuss the following results.

- **Synthetic mobility traces:** In addition to RW node mobility (`T1`), we also employ *random way-point* (RWP) mobility [42], which we refer as `T2`. In RWP, every α-node pauses for a time chosen randomly from range $[3, 8]s$, picks a random destination, and then moves there with a random velocity within $[v_{min}, v_{max}] = [1, 10]$ *m/s*. These steps are repeated until the network lifetime ends. RWP (`T2`), together with RW (`T1`), represent the most prevalent synthetic traces in network analysis.

  To evaluate the average packet propagation speed, we divide the network area into cells with side-length $l = 500m$, which is significantly larger than the transmission range of $r = 25m$. We employ $n = 100$ mobile α-nodes per cell, employ time slots with 1s length, simulate up to 8.64×10^5 slots, and generate 4×10^4 network realizations using OmNeT++'s Inetmanet-2.0 framework. We divide the cell area into rings $\mathcal{R} = \{(a_i, b_i]\}$ with $\gamma = b_i - a_i = 5m$ thickness, and plot the average speed for the i^{th} bin, $E(v_t(i))$, in Fig. 2.5b. Let the *overall average speed* be denoted as

$$\bar{v} := \frac{1}{|\mathcal{R}|} \sum_{i=0}^{|\mathcal{R}|-1} E(v_t(i)). \tag{2.9}$$

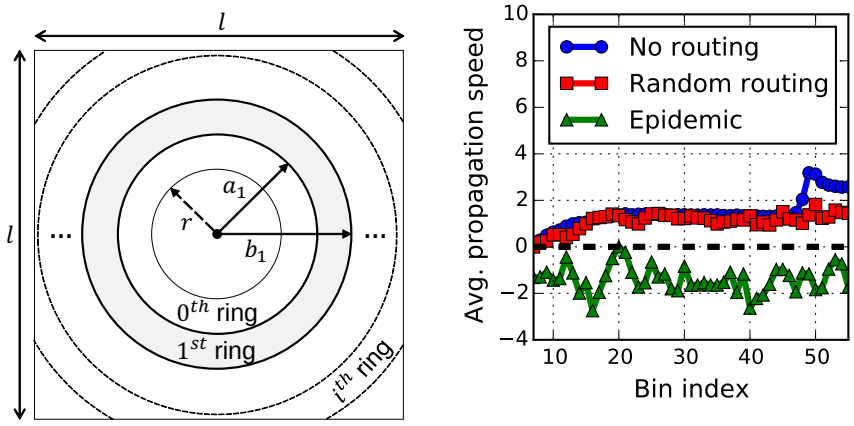
  Then, according to the results presented in Table 2.1, we observe the following result: when the overall speed is $\bar{v} > 0$ (resp., $\bar{v} < 0$), then the corresponding speed for each bin—$\forall i, E(v_t(i))$—in Fig. 2.5b is also larger (resp., smaller) than zero. For example, RW with direct contact routing (`T1+R1`) has $\bar{v} > 0$, indicating a submartingale-like packet mobility that matches the analytical result in Lemma 1. When randomized routing is used (`T1+R2`), $\bar{v}$ *decreases* but is still larger than zero. On the other hand, $\bar{v} < 0$ when epidemic routing is used (`T1+R3`), indicating a supermartingale-like mobility in which packet ϵ tends to move closer to the cloudlet β-node.

  According to Table 2.1, similar results as above can be observed when RWP (`T2`) is employed. However, it is known that the α-nodes in RWP tend to move towards the center-located β-node [30], which results in a smaller expected propagation speed $\bar{v}$.

- **Realistic mobility traces:** In addition to synthetic node mobility traces (`T1` and `T2`), which are mainly based on simplified assumptions, we also consider measurement-based mobility traces from real-world scenarios. Specifically, we choose the following mobile traces to represent disaster-response scenarios in public spaces (i.e., amusement parks).

  `T3)` `ncsu/statefair` [**126**]: Real-world GPS-based trace from 19 visitors of two North Carolina State fair events from August 26, 2006 to April 18, 2008.

  `T4)` `ncsu/disneyworld` [**126**]: GPS traces of 41 volunteers that visited Disney World, Florida during Thanksgiving and Christmas holidays in 2006 and 2007, respectively.



(a) Rings for calculating $E(v_t(i))$. (b) Result for random walk mobility.

Figure 2.5: Evaluation of supermartingale, martingale, and submartingale-like packet mobility properties.

Table 2.1: Expected propagation speed $\bar{v}$ of synthetic traces.

Node Mobility	Routing Scheme	$\bar{v}[m/s]$	Packet Mobility Region
Random Walk (T1)	Direct contact (R1)	0.10266	Submartingale
	Randomized (R2)	0.08675	
	Epidemic (R3)	-0.04357	Supermartingale
Random Waypoint (T2)	Direct contact (R1)	0.09345	Submartingale
	Randomized (R2)	0.08892	
	Epidemic (R3)	-0.04599	Supermartingale

Further, we divide the large network area, which is $(1,000 \times 1,000)m^2$ in T3 and $(18,000 \times 18,000)m^2$ in T4, into cells with $l = 500m$ side-length, while re-using the same network parameters for generating the results in Table 2.1. The corresponding overall speeds are presented in Table 2.2. In general, similar results as for the synthetic RW and RWP mobilities are observed: (i) Realistic mobility traces with direct contact (R1) and randomized (R2) routings result in submartingale-like packet mobilities, while (ii) the ones employing epidemic routing (R3) exhibit supermartingale-like properties.

The results in and Tables 2.1-2.2, combined with Lemma 1 and Corollary 1, show that the examined packet mobilities for cloudlet-based IoT in challenging environments exhibit either supermartingale, martingale, or submartingale-like properties. In the following section, we will derive the bounds of expected CAD for these three packet mobility cases, by utilizing the average speed, $\bar{v}$, obtained in Tables 2.1 and 2.2.

Table 2.2: Expected propagation speed $\bar{v}$ of realistic traces.

Node Mobility	Routing Scheme	$\bar{v}[m/s]$	Packet Mobility Region
`ncsu/statefair` (`T3`)	Direct contact	0.43387	Submartingale
	Randomized	0.41572	
	Epidemic	-0.46749	Supermartingale
`ncsu/disneyworld` (`T4`)	Direct contact	2.00356	Submartingale
	Randomized	1.97402	
	Epidemic	-0.40107	Supermartingale

2.5 Analysis of Cloudlet Access Delay

We showed that IoT applications in challenging environments have either a supermartingale, martingale, or submartingale-like packet mobility. In this section, we analyze the bounds and scaling order of CAD for these three mobility cases when the network is served by a single cloudlet with a cell area $l \times l$ (see Section 2.3.2) with an infinite side length of $l \to \infty$. Then, we examine how the CAD will be affected under multiple cloudlets with bounded cell areas.

2.5.1 Cloudlet Access Delay Under Unbounded Cell Coverage

When packet ϵ is generated by an IoT device at a location close to the nearest cloudlet, it is highly possible that ϵ will reach the nearest β-node before hitting the cell boundary. In this case, the effect of the finite boundary to the upper bound of $L_\epsilon(t)$ is insignificant and may be neglected, which is equivalent to having D2D-Cloudlet with an unrestricted cell area. In this subsection, *we analyze the expected CAD, denoted as $E(T_{CAD})$, of supermartingale, martingale, and submartingale-like packet mobility by assuming that the cell area is unbounded.* We will show in Section 2.5.2 that, depending on the initial distance-to-destination $L_\epsilon(0)$ and the cell side length l, the analyses also hold when the cell is bounded.

In view of delay analysis, the latency of hitting a target in the middle of a circular network area with radius a and cloudlet transmission range of b is shown to be $\Theta(\frac{a^2}{b})$ under random direction forwarding and $\Theta(a)$ with location-based routing [83]. Unlike the specific combinations of node mobility and forwarding schemes considered in [83], however, we assume a generic packet mobility scenario. Moreover, in contrast to [83] in which the latency is an average over all possible initial locations, our CAD analysis corresponds to an exact initial distance-to-destination, $L_\epsilon(0)$. Thus, the corresponding expected CAD, $E(T_{CAD})$, presented below is more general in the sense that the results in [83] is obtainable through averaging $E(T_{CAD})$ over all possible $L_\epsilon(0)$'s.

To proceed, we outline the following CAD results for the three distinct packet mobility

cases.

Supermartingale-like packet mobility

First, we examine the case of packet mobility with supermartingale-like property. Since packet ϵ has a long-term tendency to move closer towards β-node, our intuition suggests that the expected delay $E(T_{CAD})$ should be *smaller* than the other packet mobility cases. We justify this by deriving the lower and upper bounds of $E(T_{CAD})$ and show that the access delay scales *linearly* with respect to the initial distance-to-destination $L_\epsilon(0)$, which grows slower than the other packet mobility cases.

To analyze the bounds of the expected CAD, we will use *Doob's decomposition* [90, Thm. 25.3] to build a new martingale sequence $\{L'_\epsilon(t)\}$ out of the supermartingale $\{L_\epsilon(t)\}$, apply the *optional stopping theorem* (OST) [90, Thm. 24.6] to predict the value of the martingale sequence at packet delivery (i.e., $L'_\epsilon(T_{CAD})$), and then use Wald's first inequality to obtain the bounds in (2.10). The main challenge in our analysis is to verify the expected CAD's boundedness ($E(T_{CAD}) < \infty$) as a necessary condition for using the OST, and showing that $\{L_\epsilon(t)\}$ is IID before applying Wald's inequality.

Theorem 1. *For independently-distributed packet propagation speeds $\{v_t\}$ and a small interval gap of $\gamma = (b_i - a_i) \to 0$, the expected cloudlet access delay for supermartingale-like packet mobility is bounded by*

$$\frac{L_\epsilon(0) - r}{-\bar{v}} \leq E(T_{CAD}) \leq \frac{L_\epsilon(0)}{-\bar{v}}. \tag{2.10}$$

Proof. To prove Theorem 1, we define $L'_\epsilon(t) := L_\epsilon(t) - H_t$, where $H_t := \sum_{k=1}^{t} E(v_k|\mathcal{F}_{k-1})$. When $\gamma \to 0$, $\{L_\epsilon(t) : t \in \mathcal{T}\}$ becomes a supermartingale and we have

$$E(L'_\epsilon(t)|\mathcal{F}_{t-1}) = E(L_\epsilon(t) - v_t - H_{t-1}|\mathcal{F}_{t-1}) = L_\epsilon(t-1) - H_{t-1} = L'_\epsilon(t-1), \tag{2.11}$$

which implies that the process $\{L'_\epsilon(t) : t \in \mathcal{T}\}$ is a martingale. We will use this property to derive the lower and upper bounds of the expected cloudlet access delay, $E(T_{CAD})$.

We notice that T_{CAD} is a *stopping time* [75, 90] since it depends on the packet ϵ's location up to the current time step (not the future) and is $\mathcal{F}_t$-measurable. In the literature, Doob's optional stopping theorem (OST) [90] is a powerful tool for analyzing the stopping time until a gambler's fortune is ruined [75] and until a random walker returns to origin or hits an upper bound [23]. Applying the OST, however, requires the stopping time to be finite, which is not necessarily true in our IoT-based network since the packet propagation speeds are random. Thus, we need to check the boundedness of T_{CAD} prior to adapting the OST.

Let $\{v_t\}$ be *independent* random variables. Such an assumption holds when the packet's location at the current time t—and the corresponding $L_\epsilon(t)$—is independent to the locations at the previous time slots (captured by $\{L_\epsilon(k)\}_{k<t}$), e.g., under RW node mobility without routing. On the other hand, when routing is employed under a more general node mobility model, packet ϵ is constantly copied or forwarded to the newly-contacted nodes. The positions of these contacted nodes are known to be uncorrelated to the locations of the packet-carrying α-nodes from the previous time slots [25, Sec. 2]. Thus, the independent assumption regarding $\{v_t\}$ is also valid when routing is employed, since there is a high chance of contacting a new α-node at every time slot, especially as the number of nodes n increases.

To proceed, let $T' := \inf\{t : L_\epsilon(0) + \sum_{k=1}^{t} v_k = 0\}$ be the time until a packet ϵ hits the origin point, $\vec{X}_\beta$. Clearly, $T_{CAD} < T'$. Since $\{v_t\}$ are independent and, by assumption, identically-distributed, then Wald's first equality states that $E(T_{CAD}) < E(T') = \frac{L_\epsilon(0)}{-E(v_t|\mathcal{F}_{t-1})} < \infty$. This inequality, combined with the fact that $E(|L'_\epsilon(t) - L'_\epsilon(t-1)||\mathcal{F}_{t-1}) = E(L'_\epsilon(t) - L'_\epsilon(t-1)|\mathcal{F}_{t-1}) = 0$, indicate that the two conditions for applying the OST [75] are satisfied. Accordingly,

$$\begin{aligned} E(L'_\epsilon(T_{CAD})|\mathcal{F}_0) = L'_\epsilon(0) \quad &\Leftrightarrow E((L_\epsilon(T_{CAD}) - H_{T_{CAD}}) - (L_\epsilon(0) - H_0)) = 0 \\ &\Leftrightarrow E(L_\epsilon(0) - L_\epsilon(T_{CAD})) = -H_{T_{CAD}}. \end{aligned} \tag{2.12}$$

Upon delivery, packet ϵ is already within the transmission range of a β-node, that is, $0 \leq L_\epsilon(T_{CAD}) \leq r$, which can be re-formulated into

$$L_\epsilon(0) - r \leq -H_{T_{CAD}} \leq L_\epsilon(0). \tag{2.13}$$

Since the packet propagation speed v_t for all $t \in \mathcal{T}$ are independent and identically-distributed (IID), then $\bar{v} = E_{\mathcal{F}_{t-1}}(E(v_t|\mathcal{F}_{t-1}))$ and $E(H_{T_{CAD}}) = \bar{v}E(T_{CAD})$ according to Wald's equation. Plugging this result into the expected value of (2.13) obtains Theorem 1. □

Based on Theorem 1, we can analyze the scaling law of $E(T_{CAD})$ with respect to the initial distance-to-cloudlet $L_\epsilon(0)$. To do so, we present the following definition according to [52].

Definition 7 (Scaling law)**.** *We say: (i) $f(x) = O(g(x))$ if there exists a constant $c > 0$ such that $f(x) \leq cg(x)$ as $x \to \infty$, (ii) $f(x) = \Omega(g(x))$ if $g(x) = O(f(x))$, and (iii) $f(x) = \Theta(g(x))$ if $f(x) \leq cg(x)$ and $f(x) = \Omega(g(x))$.*

Then, the following corollary can be obtained by applying the definition above to (2.10).

Corollary 2. *For a supermartingale-like packet mobility, the expected delay scales with the order of $E(T_{CAD}) = \Theta(L_\epsilon(0))$.*

Such a result indicates that the expected delay $E(T_{CAD})$ increases *linearly* versus the initial distance-to-destination $L_\epsilon(0)$.

Submartingale-like packet mobility

For IoT-based applications in challenging environments with submartingale-like mobility, packet ϵ have the long-term tendency to move away from β-node. As a result, we expect that the packet requires a longer delay to be delivered, which results in a higher expected access delay $E(T_{CAD})$, compared to the supermartingale-like case. To justify this, we first derive the lower and upper bounds of $E(T_{CAD})$ in Theorem 2. In this case, packet ϵ tends to move away from the cloudlet β-node and it is hard to check whether T_{CAD} is finitely bounded from above. As a result, the necessary condition for applying the OST is not satisfied and the steps of proof in Theorem 1 cannot be re-applied here. Instead we derive the bounds of $E(T_{CAD})$ via algebraic manipulations, to get the following theorem.

Theorem 2. *Let v_{max} be the maximum velocity of the α-nodes and define $\bar{\sigma}^2 := E_t(\sigma_t^2)$, where $\sigma_t^2 := E(v_t^2|\mathcal{F}_{t-1}) - E^2(v_t|\mathcal{F}_{t-1})$. For identically-distributed $\{v_t\}$, the expected access delay of a D2D-Cloudlet with a submartingale-like packet mobility satisfies*

$$\frac{(L_\epsilon(0)-r)^2}{\bar{\sigma}^2+\bar{v}^2+2(v_{max}+r)} \leq E(T_{CAD}) \leq \frac{(L_\epsilon(0))^2}{\bar{\sigma}^2+\bar{v}^2}. \tag{2.14}$$

Proof. From Section 2.3.2, we know that $L_\epsilon(t) = L_\epsilon(0) + \sum_{i=1}^{t} v_i$, which can be restated as

$$\begin{aligned}\sum_{i=1}^{t} v_i^2 =& (L_\epsilon(t) - L_\epsilon(0))^2 - 2\sum\nolimits_{1\leq i<j\leq t} v_i v_j \\ =& (L_\epsilon(t) - L_\epsilon(0))^2 - 2\sum_{i=2}^{t} v_i L_\epsilon(i-1) + 2\sum_{i=2}^{t} v_i L_\epsilon(0).\end{aligned} \tag{2.15}$$

Consider the packet delivery instance of T_{CAD}. After plugging T_{CAD} into (2.14) and applying an expectation operator $E(\cdot)$ to both sides of the equation, we obtain

$$\sum_{i=1}^{T} E(v_i^2|\mathcal{F}_{i-1}) = E\left((L_\epsilon(T_{CAD}) - L_\epsilon(0))^2\right) - 2\sum_{i=2}^{T} E(v_i|\mathcal{F}_{i-1})\left(L_\epsilon(i-1) - L_\epsilon(0)\right) \tag{2.16}$$

When $\gamma \to 0$ in the submartingale-like packet mobility case, $E(v_t|\mathcal{F}_{t-1}) \geq 0$ and $L_\epsilon(0) \geq 0$ for all $t \in \mathcal{T}$, such that

$$\begin{aligned}E\left((L_\epsilon(0) - L_\epsilon(T_{CAD}))^2\right) \leq& \sum_{i=1}^{T_{CAD}} \left[E(v_i^2|\mathcal{F}_{i-1}) + 2(v_{max}+r)E(v_i|\mathcal{F}_{i-1})\right] \\ =& E(T_{CAD})\left[E(v_t^2|\mathcal{F}_{t-1}) + 2(v_{max}+r)E(v_t|\mathcal{F}_{t-1})\right] \\ =& E(T_{CAD})\left[\bar{\sigma}^2 + (\bar{v})^2 + 2(v_{max}+r)\bar{v}\right].\end{aligned} \tag{2.17}$$

Moreover, the distance traversed by ϵ during two consecutive time slots is upper bounded by the maximum speed v_{max} due to packet carriage plus the maximum forwarding distance r. Thus,

(2.17) can be obtained by implementing another expectation to the line above it, applying the triangle inequality $v_t \leq v_{max}+r$, and employing Wald's first equality for IID packet propagation speeds. Accordingly, the lower bound in (2.14) can be obtained because $L_\epsilon(T_{CAD}) \leq r$, i.e., the packet is already within β-node's transmission range r at delivery.

To prove the right hand of (2.14), we re-state (2.15) as follows.

$$E\left((L_\epsilon(0) - L_\epsilon(T_{CAD}))^2\right) \geq E(T_{CAD}) \times E(v_t^2|\mathcal{F}_{t-1}) = E(T_{CAD})\left[\bar{\sigma}^2 + (\bar{v})^2\right]. \quad (2.18)$$

Finally, the upper bound in (2.14) can be obtained by plugging $L_\epsilon(T_{CAD}) \geq 0$ into (2.18). □

Accordingly, the following corollary can be obtained by applying Definition 7 to Theorem 2.

Corollary 3. *In submartingale-like mobility, the expected access delay scales in the order of* $E(T_{CAD}) = \Theta(L_\epsilon^2(0))$.

We can observe that the expected access delay of a submartingale-like process $\{L_\epsilon(t)\}$ grows with a *quadratic* factor of the initial distance-to-destination, *faster* than the linear growth of a supermartingale-like packet mobility.

Martingale-like packet mobility

In this case, we observe that $E(v_t|\mathcal{F}_{t-1}) = 0$, which serves as the borderline between the supermartingale-like and the submartingale-like regimes. One open question comes into our mind: whether martingale-like packet mobility exhibits an access delay that is similar to the former or the latter. We answer this question through the following theorem, which can be obtained after following the similar steps as in the proof of Theorem 1. Instead of dealing with the supermartingale-like case with a packet movement tendency to move towards the cloudlet over time, however, we consider the setting where ϵ tends to stay at the same distance-to-destination.

Theorem 3. *The expected access delay of a D2D-Cloudlet with a martingale-like packet mobility satisfies*

$$\frac{(L_\epsilon(0) - r)^2}{\bar{\sigma}^2} \leq E(T_{CAD}) \leq \frac{(L_\epsilon(0))^2}{\bar{\sigma}^2}. \quad (2.19)$$

Proof. To prove Theorem 3, denote $\{Y_t : t \in \mathcal{T}\}$ as an auxiliary process, where $Y_t := (L_\epsilon(0) - L_\epsilon(t))^2 - S_t$ and $S_t := \sum_{k=1}^{t} \sigma_k^2$. Then, we get

$$\begin{aligned} E(Y_t|\mathcal{F}_{t-1}) &= E([(L_\epsilon(0) - L_\epsilon(t))^2 - S_t]|\mathcal{F}_{t-1}) \\ &= E([(L_\epsilon(0) - L_\epsilon(t-1))^2 - S_{t-1} - 2v_t(L_\epsilon(0) - L_\epsilon(t-1)) + v_t^2 - \sigma_t^2]|\mathcal{F}_{t-1}). \end{aligned}$$

For a martingale-like packet mobility when $\gamma = b_i - a_i \to 0$ ($a_i \leq L_\epsilon(t-1) < b_i$), we have $E(L_\epsilon(0) - L_\epsilon(t-1)|\mathcal{F}_{t-1}) = 0$ and $\sigma_t^2 = E(v_t^2)$. As a result, (2.20) can be re-stated as

$$E(Y_t|\mathcal{F}_{t-1}) = E(((L_\epsilon(0) - L_\epsilon(t-1))^2 - S_{t-1})|\mathcal{F}_{t-1}) = Y_{t-1}, \tag{2.20}$$

which indicates $\{Y_t\}$ is also a martingale. We will use this property to derive the expected cloudlet access delay, $E(T_{CAD})$, using Doob's OST [90]. Similar to the proof of Theorem 1, the main challenge here is to check the conditions for the OST to hold, namely: (i) $E(T_{CAD}) < \infty$, and (ii) $E(|Y_t - Y_{t-1}||\mathcal{F}_{t-1}) \leq c$ for all $t \in \mathcal{T}$ and some constant c.

To proceed, we re-employ the time to hit the origin, T', as in Appendix **??** and apply Wald's second equality to get $E(T_{CAD}) \leq E(T') = \frac{E(L_\epsilon(0))^2}{\bar{\sigma}^2} < \infty$, which satisfies the first condition mentioned above. To check the second condition, we notice that $|Y_t - Y_{t-1}| = |L_\epsilon^2(t)| - 2|L_\epsilon(0)(L_\epsilon(t) - L_\epsilon(t-1))| - |L_\epsilon^2(t-1)| - |\sigma_t^2|$ because $L_\epsilon(t)$ and σ_t^2 are positive for all $t \in \mathcal{T}$. The highest value of $E(|Y_t - Y_{t-1}||\mathcal{F}_{t-1})$ is when the packet ϵ moves from the corner of the cell towards the cloudlet β-node with a maximum movement propagation distance of $r + v_{max}$. In this case,

$$E(|Y_t - Y_{t-1}||\mathcal{F}_{t-1}) = E(|L_\epsilon^2(t)| - 2|L_\epsilon(0)(L_\epsilon(t) - L_\epsilon(t-1))| - |L_\epsilon^2(t-1)| - |\sigma_t^2||\mathcal{F}_{t-1}) \tag{2.21}$$

$$\leq E(|L_\epsilon^2(t) - L_\epsilon^2(t-1)||\mathcal{F}_{t-1}) \tag{2.22}$$

$$\leq (\sqrt{2}l - (r + v_{max}))(r + v_{max}), \tag{2.23}$$

which indicates the second condition also holds since the cell area of a cloudlet is much larger than the maximum packets movement distance (i.e., $l > (r + v_{max})$). Since both conditions (i) and (ii) are satisfied and $\{v_t\}$ are martingale, we can employ the OST that yields

$$E((L_\epsilon(0) - L_\epsilon(T_{CAD}))^2) = S_{T_{CAD}}. \tag{2.24}$$

On the other hand, according to our definition in (2.3), the distance-to-destination when a packet is delivered satisfies $0 \leq L_\epsilon(T_{CAD}) \leq r$, which can be combined with (2.24) to obtain

$$(L_\epsilon(0) - r)^2 \leq S_{T_{CAD}} \leq (L_\epsilon(0))^2. \tag{2.25}$$

Because the packet propagation speeds $\{v_t\}$ are IID, then $E(S_{T_{CAD}}) = E(\sum_{k=1}^{T_{CAD}} \sigma_k^2) = \bar{\sigma}^2 E(T_{CAD})$. This equation can be substituted into (2.25) to get Theorem 3. □

Corollary 4. *A martingale-like mobility has an expected access delay with a scaling order of* $E(T_{CAD}) = \Theta(L_\epsilon^2(0))$.

Such a corollary proves that a martingale-like packet mobility has an $E(T_{CAD})$ that scales

quadratically versus $L_\epsilon(0)$, resembling the result for the submartingale-like process.

Remark 1. *Corollaries 2-4 suggest that the best (slowest-growing) expected access delay, $E(T_{CAD})$, is achieved by supermartingale-like packet mobility. In this case, according to Theorem 1, the lowest $E(T_{CAD})$ is when the expected packet propagation speed $\bar{v}$ is minimized.*

2.5.2 Effect of Finite Cell Boundary to Cloudlet Access Delay

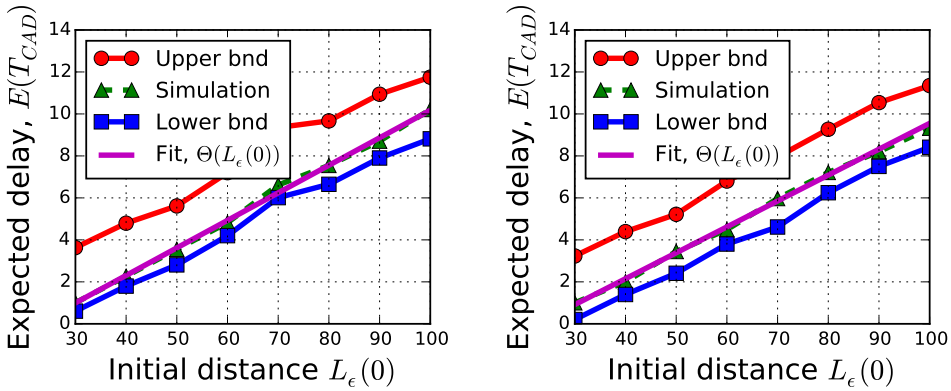
After deriving the bounds and scaling laws of $E(T_{CAD})$ for networks with infinite cell area, we return to a more practical setting by considering multiple cloudlets, each serving a cell area with *bounded* side-length, i.e., $l \ll \infty$. Specifically, we compare the scaling order of $E(T_{CAD})$ under bounded cell area to that with unbounded cell size, which is studied in the previous subsection.

The comparison between bounded and unbounded networks has been done in terms of their mobile nodes' inter-meeting time distributions. For instance, the distribution is found to be exponential when the cell size is finite; otherwise, it follows a power-law [41]. While Markovian models can evaluate the end-to-end delay by taking as input the exponential inter-meeting times [174, 76, 119], the analysis for more general distributions (i.e., the power-law case) is still lacking. Moreover, existing end-to-end delay analyses via inter-meeting times are unsuitable for IoT-based networks where delay (CAD) is measured from mobile-to-destination and is mainly a function of the initial distance, $L_\epsilon(0)$.

Motivated by the lack of study, we examine how the finite cell boundaries will affect the expected CAD, $E(T_{CAD})$. To do this, we denote *boundary effect* as the difference between the expected CAD of the IoT networks with bounded cell area to that of the unbounded one, under the same node mobility and forwarding schemes. For example, if the expected delay of the bounded and unbounded cases are 2s and 4s, respectively, then the boundary effect reduces $E(T_{CAD})$ by two seconds. Let the severity of the boundary effect be defined as follows.

Definition 8. *For IoT-applied networks with finite cell side-length, the boundary effect is non-significant if the corresponding scaling law analyses in Corollary 2, 3, or 4 still holds. Otherwise, the boundary effect is said to be significant.*

With Definition 8, we are ready to analyze the boundary effect in supermartingale, submartingale, and martingale-like packet mobility cases. To this end, we perform numerical simulations in OmNeT++, with a simulation update interval of 1s. The network is divided into cells with $l \times l = (10^3 m)^2$ areas, in which there are $n = 100$ α-nodes per cell. We use the carrier frequency of 2.4GHz, while the medium access control (MAC)-layer bit rate is set to 2Mbps. The transmission power is 2mW and the signal attenuation threshold is -66dBm, while the fading model is a path-loss model with a coefficient of 2, which correspond to the node transmission range of $r = 25m$.



(a) Cell side length $l = 1000m$.

(b) Cell side length $l = 500m$.

Figure 2.6: Expected access delay for supermartingale-like packet mobility.

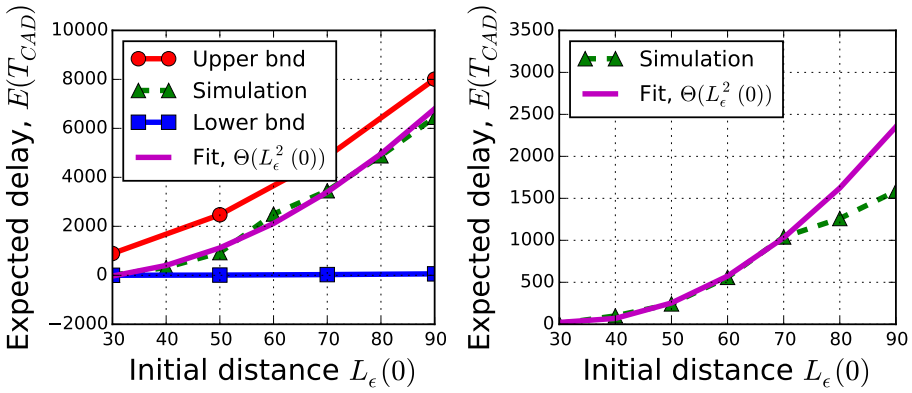
Supermartingale-like packet mobility

To evaluate the boundary effect, we consider the result in Corollary 2 that describes the scaling order of $E(T_{CAD})$ versus the initial distance-to-destination, $L_\epsilon(0)$. Evaluating the scaling order requires other network parameters, including the overall propagation speed $\bar{v}$, to be fixed. Accordingly, we employ the RWP node mobility and epidemic routing (`T2+R3`) that has $\bar{v} \approx -4.342 \times 10^{-2} m/s$ and fits this requirement.

In such a supermartingale case, recall that packet ϵ tends to move towards the cloudlet (β-node) over time. Consequently, it is likely that ϵ can arrive at the β-node before hitting the cell boundary, in which the effect of finite boundary should be negligible. To verify this, we present the expected access delay $E(T_{CAD})$ from numerical simulations as the dotted green plot in Fig. 2.6a, and then compare it to the analyzed lower and upper bounds (see the solid blue and red plots). The resulting plots show that the bounds in (2.10) are valid, while the scaling order in Corollary 2 also holds, as verified by the linear fit [152] in Fig. 2.6a, indicating the boundary effect is insignificant. Moreover, the observations regarding the bounds and scaling laws also hold when the cell side length is reduced to $l = 500m$ (see Fig. 2.6b).

Submartingale-like packet mobility

In this case, the packet tends to move away from the cloudlet towards the finite cell boundary. Since boundary reflection mapping is assumed (see Section 2.3.2), then every packet movement across the cell border will be mirrored back into the original cell, which effectively brings ϵ closer to the β-node and diminishes the access delay. Specifically, an ϵ that begins near the border is more likely to hit the border, causing a more dramatic reduction in $E(T_{CAD})$, versus



(a) Cell side length $l = 1000m$. (b) Cell side length $l = 500m$.

Figure 2.7: Expected access delay for submartingale-like packet mobility.

when ϵ starts near the β-node.

To validate such a conjecture, we plot the expected access delay $E(T_{CAD})$ of a random walk mobility with a randomized routing (`T1+R2`) in Fig. 2.7a. This packet mobility case is selected since it exhibits a relatively constant expected speed $\bar{v}$ and variance $\bar{\sigma}^2$, which according to the bounds in (2.14) should give a clear indication on how $E(T_{CAD})$ will grow versus the initial distance $L_\epsilon(0)$. From the figure, we observe that the $E(T_{CAD})$ agrees with its lower and upper bounds, and exhibits a quadratic scaling law as predicted in Corollary 3.

When the cell side length is reduced to $l = 500m$, on the other hand, the expected delay becomes smaller and the quadratic scaling cannot be observed for all the examined $L_\epsilon(0)$'s anymore (see Fig. 2.7b). Such an observation holds because when the initial distance is large (e.g., $L_\epsilon(0) \geq 80m$), the packet starts near the cell boundary and has a non-negligible probability of hitting the boundary before being delivered. This verifies (i) the boundary effect is *insignificant* for large l, and (ii) it becomes *significant* when l is small compared to the initial distance $L_\epsilon(0)$.

Martingale-like packet mobility

We employ an Infostation architecture [67] that serves a two-way road, in which every vehicular α-node has a uniformly-distributed velocity v_{node} within the range of $[1, 9]m/s$. Moreover, we employ randomized routing (`R2`), which according to Lemma 1 should result in a packet mobility with martingale properties. Based on the expected access delay plot in Fig. 2.8, we observe that, despite a very large side length of $l = 10^4m$, the quadratic scaling in Corollary 4 can only be observed up to the initial distance of $L_\epsilon(0) = 90m$. Beyond this initial distance, the boundary effect becomes *significant* and the scaling law cannot be observed anymore.

Remark 2. *Figures 2.7 and 2.8 indicate that $E(T_{CAD})$ becomes lower when the boundary effect*

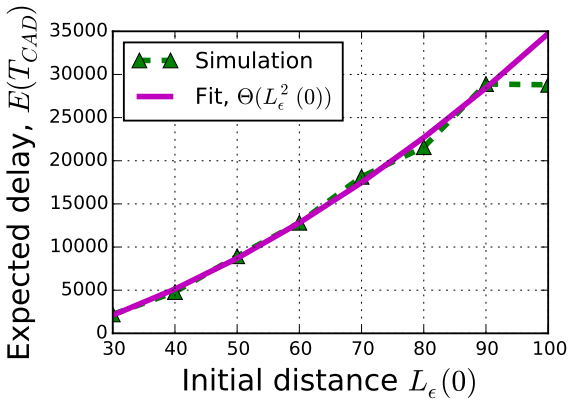


Figure 2.8: Expected access delay for martingale-like packet mobility.

is significant (such as when $L_\epsilon(0) \geq 90m$ in Figs. 2.7b and 2.8). Thus, the expected delay is improved under significant boundary effect.

So far, we have analyzed the bounds of CAD as well as how it is affected by finite boundary. In the following, we will show how to achieve satisfactory CAD while fulfilling our second main objective: to minimize the per-message power draw P_ϵ.

2.6 Fast and Efficient Packet Delivery

Many IoT applications in challenging environments are geared toward providing time-critical emergency services. For example, search-and-rescue (SAR) in disaster areas is a constant battle against time. The sooner a victim is located and rescued, the more likely his/her death can be avoided. Motivated by the importance, we first analyze the IoT network's ability to support strict delay requirements in Section 2.6.1. Then, we show how to fulfill the time-critical delay requirement while minimizing the power consumed by IoT devices in Section 2.6.2.

2.6.1 Satisfying Time Critical Requirement

In order to support time-critical applications, a packet ϵ should arrive at a cloudlet within a finite delay constraint of τ time slots (recall Section 2.3.3). When the delay constraint τ is exceeded, packet ϵ will be useless and need to be discarded. To quantify the network's ability to deliver packet within such a constraint, let *successful access rate* $\Pr\{T_{CAD} < \tau\}$ be the likelihood that the cloudlet can be accessed successfully within the required delay. A large $\Pr\{T_{CAD} < \tau\}$ means packet ϵ is more likely to be delivered in time, and time-critical applications can be supported better. Thus, our aim is to maximize $\Pr\{T_{CAD} < \tau\}$.

In the literature, the impacts of the number of nodes and transmission range to $\Pr\{T_{CAD} < \tau\}$ for greedy location-based routing in DTN is numerically studied in [134]. On the other hand, Shirani *et al.* employed quadratic curve-fitting for studying how $\Pr\{T_{CAD} < \tau\}$ is affected by the number of packet hops [135]. The aforementioned studies, however, assume a specific forwarding scheme—which may not be extendable to other packet mobility scenarios—and are focused on the mobile-to-mobile successful access rate, unlike the mobile-to-cloudlet setup considered in this chapter. Moreover, [134] and [135] neglects the packet propagation speed $\bar{v}$ in their analysis.

Motivated by the lack of suitable analytical result, we derive a generic relation between $\Pr\{T_{CAD} < \tau\}$ and $\bar{v}$. We start by presenting the bounds of successful access rate in Corollary 5 below. The corollary is obtained by fixing the delay constraint τ, and then calculating the likelihood that packet ϵ is already within the range of a cloudlet (i.e., $L_\epsilon(\tau) < r$) using concentration inequalities [50]. We get the following theorem.

Corollary 5. *Denote $M := r + v_{max}$ as the maximum distance that can be traveled by a packet at each time step and $[x]^+ := \max(0, x)$. Let the packet propagation speeds, $\{v_t\}$, be identically-distributed. Then, the successful rate satisfies*

$$\begin{aligned} &1 - \exp\left\{ -\frac{\left([r - L_\epsilon(0) - \tau\bar{v}]^+\right)^2}{2\left(\tau(\bar{\sigma}^2 + (\bar{v})^2) + \frac{M}{3}\left[r - L_\epsilon(0) - \tau\bar{v}\right]^+\right)} \right\} \\ &\qquad \leq \Pr\{T_{CAD} < \tau\} \\ &\leq \exp\left\{ -\frac{\left([\tau\bar{v} + L_\epsilon(0) - r]^+\right)^2}{2\left(\tau(\bar{\sigma}^2 + (\bar{v})^2) + \frac{M}{3}\left[\tau\bar{v} + L_\epsilon(0) - r\right]^+\right)} \right\}. \end{aligned} \tag{2.26}$$

Proof. To prove the left-hand side of (2.26), we notice that the distance travelled by packet ϵ at each time slot is upper bounded by $|v_t| \geq M$ where $M = r + v_{max}$. If ϵ has been delivered within a delay τ, the packet is already within the transmission radius r of a β-node, that is, $L_\epsilon(\tau) \leq r$. By applying concentration inequality [50, Thm. 6], the probability that ϵ has not been delivered at time τ can be stated as

$$\Pr\{T \geq \tau\} = \Pr(L_\epsilon(\tau) \geq r) \leq \exp\left(-\frac{\lambda_1^2}{2(\xi_\tau^2 + M\lambda_1/3)}\right), \tag{2.27}$$

where

$$\lambda_1 = [r - L_\epsilon(0) - \sum_{t=1}^{\tau} E(v_t|\mathcal{F}_{t-1})]^+ = [r - L_\epsilon(0) - \tau\bar{v}]^+, \tag{2.28}$$

$$\text{and } \xi_\tau^2 = \sum\nolimits_{i=1}^{\tau} E(v_i^2|\mathcal{F}_{i-1}) = \tau(\bar{\sigma}^2 + (\bar{v})^2), \tag{2.29}$$

because $\bar{v} = E_t(E(v_t|\{_{t-1})) $ and $\bar{\sigma}^2 = E_t(E(v_t^2|\mathcal{F}_{t-1})) - E_t(E(v_t|\mathcal{F}_{t-1}))^2$ when $\{v_t\}$ are IID.

The lower bound in (2.26) can then be obtained by taking the complementary probability of (2.27).

To prove the right-hand side of (2.26), suppose $v_t \geq -M$ for all $t \in \mathcal{T}$. Then, the packet delivery upper bound can be calculated using [50]

$$\Pr\{T \geq \tau\} = \Pr\{L_\epsilon(\tau) \leq r) \leq \exp\left(-\frac{\lambda_2^2}{2(\xi_\tau^2 + M\lambda_2/3)}\right), \tag{2.30}$$

where $\lambda_2 := [\sum_{i=1}^{\tau} E(v_i|\mathcal{F}_{i-1}) + L_\epsilon(0) - r]^+ = [\tau\bar{v} + L_\epsilon(0) - r]^+$. Finally, the right-hand side of (2.26) can be obtained by combining (2.28), (2.29), and λ_2 into (2.30). □

Corollary 5 is general and does not assume any specific packet mobility setup, as long as the packet mobility is either supermartingale, martingale, or submartingale-like. The corollary, however, does not provide a clear relation between $\Pr\{T_{CAD} < \tau\}$ and the average packet propagation speed $\bar{v}$. Such a relation is characterized in the following lemma.

Lemma 2. *The successful access rate scales according to*

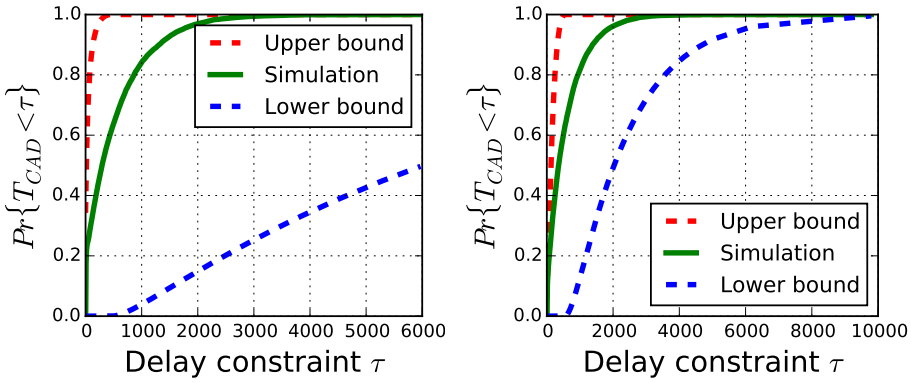
$$\Pr\{T_{CAD} < \tau\} = O(e^{-\bar{v}}). \tag{2.31}$$

Proof. From the upper bound in (2.26), we have $\Pr\{T_{CAD} < \tau\} \leq \exp\left\{-\frac{\left([\tau\bar{v}+L_\epsilon(0)-r]^+\right)^2}{2\frac{M}{3}[\tau\bar{v}+L_\epsilon(0)-r]^+}\right\} = \exp\{-\frac{3}{2M} \times [\tau\bar{v} + L_\epsilon(0) - r]^+\}$ since the negative exponential $\exp\{-x\}$ is a decreasing function of x. Omitting the constant factors (e.g., $(L_\epsilon(0) - r)$, $(r + v_{max})$, and $\frac{2M}{2}$) and applying the Big-O notation in Defition 7 completes the proof. □

Before proceeding, we evaluate our analysis in Corollary 5 by employing the RWP mobility (`T2`) and set the cell side-length, number of nodes per-cell, and transmission range to $l = 1,500m$, $n = 50$, and $r = 25m$. The simulation tools and remaining network parameters are identical to that used in Section 2.5.2, while the results are obtained over 1.5×10^4 network realizations.

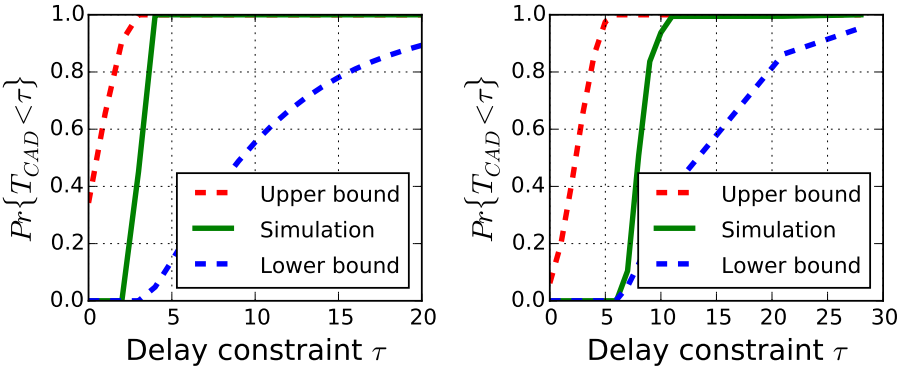
We start by considering RWP without routing (`T2+R1`) to represent the submartingale-like case. To verify Corollary 5, we set the initial distance to $L_\epsilon(0) = 50m$, plot the numerical result as a solid green line, and show that the successful access rate matches the bounds in (2.26), which are depicted as the dotted lines in Fig. 2.9a. Similar results can also be observed as the initial distance is increased to $L_\epsilon(0) = 90m$ in Fig. 2.9b.

Next, we employ RWP with epidemic routing (`T2+R3`) to represent a supermartingale-like packet mobility. From the $\Pr\{T_{CAD} < \tau\}$ plots in Fig. 2.10a, we observe that in this case the bounds in Corollary 5 are also valid. Note that the *delay spread*—the difference between the



(a) $L_\epsilon(0) = 50m$. (b) $L_\epsilon(0) = 90m$.

Figure 2.9: Successful access rate for submartingale-like packet mobility.



(a) $L_\epsilon(0) = 50m$. (b) $L_\epsilon(0) = 90m$.

Figure 2.10: Successful access rate for supermartingale-like packet mobility.

minimum and maximum access delay—is very small and the plot resembles a step function at $\tau = 2s$. Moreover, Fig. 2.10b indicates that the bounds in (2.26) become tighter as the initial distance is increased to $L_\epsilon(0) = 90m$.

Remark 3. *Figs. 2.9 and 2.10 indicate that the empirical successful access rate* $\Pr\{T_{CAD} < \tau\}$ *is close to its upper bound, which according to Lemma 2 scales as* $O(\exp\{-\bar{v}\})$. *Since* $\exp\{-\bar{v}\}$ *is a decreasing function, the highest* $\Pr\{T_{CAD} < \tau\}$ *is achievable through minimizing* $\bar{v}$.

Remark 4. *To minimize* $\bar{v}$, *researchers have proposed motion vector (MoVe) [99] and future proximity detection [21] schemes. These schemes predicts the future node location within the measurement accuracy bound of modern GPS devices [21], and constantly forward packet* ϵ *to a neighbor with the smallest distance-to-destination. As a result, the instantaneous packet propagation speed,* v_t *for all* t, *is minimized and the smallest average speed* $\bar{v}$ *can be achieved.*

Algorithm 1 Packet Shedding for Time-Critical Applications

1: **for** each packet ϵ with delay constraint τ **do**
2: calculate average speed $\bar{v} = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}\frac{r-L_{\epsilon,i}(0)}{T_{CAD,i}}$ and variance $\bar{\sigma}^2 = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I},t\leq T_{CAD}}\frac{(L_{\epsilon,i}(t)-L_{\epsilon,i}(t-1))^2-\bar{v}^2}{T_{CAD}}$.
3: determine the initial distance $L_\epsilon(0)$.
4: estimate $\hat{\Pr}\{T_{CAD}<\tau\}$ as the lower bound in (2.26).
5: **if** $\hat{\Pr}\{T_{CAD}<\tau\}<P_s$ **then**
6: deliver packet ϵ using store-carry-forwarding.
7: **else** discard ϵ.

2.6.2 Power-Efficient Packet Shedding

After analyzing the successful access rate $\Pr\{T_{CAD}<\tau\}$, we proceed to our second main objective of efficiently delivering packet ϵ to a cloudlet β-node. In particular, we exploit our analytical results in Section 2.5.1 to design a packet shedding scheme, and then show that it can minimize the power consumed during packet delivery.

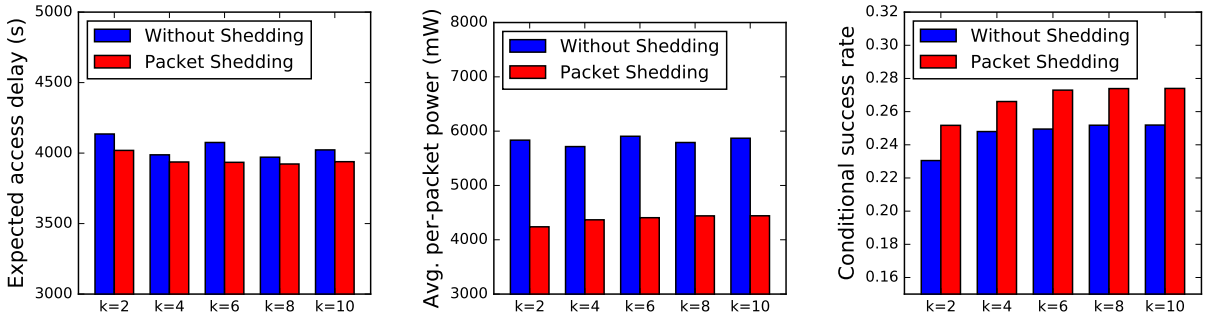
Algorithm design

Let every IoT α-device generates multiple packets, each requiring a delay constraint of τ, which are delivered to β-node using SCF. There will be two types of packets: (i) those with satisfactory delay (i.e., $T_{CAD}<\tau$); and (ii) the ones that cannot be delivered within τ time steps. Attempting to deliver the former packets is a waste of power, since these packets will arrive late, be useless, and eventually become discarded by the recipient β-node. The question is, how to determine whether a packet will belong to the second (wasteful) category.

We assume that the cloudlet β-nodes are aware of their own location, which is determinable via on-board GPS or triangulating signals from nearby devices [149], and these coordinates can be made available to nearby IoT devices via an efficient broadcast scheme [93]. After combining its own location information from in-built sensor(s) and β-nodes' positions, each α-node can determine the initial distance-to-destination $L_\epsilon(0)$ for all packets it generates. By substituting $L_\epsilon(0)$ into (2.26), the α-node can determine the successful access rate of each packet.

Let P_s be a target success rate that needs to be satisfied. When $\Pr\{T_{CAD}<\tau\}<P_s$, the α-node infers that the desired quality-of-service level cannot be satisfied and the packet will be discarded immediately to reduce unnecessary wireless transmissions. Such a packet discarding process is outlined in Algorithm 1 and is referred as a *packet shedding* scheme. The packet shedding is performed by every α-node only to the packets it generates, but not to packets copied/forwarded from nearby α-nodes.

The proposed packet shedding scheme can be described as follows. Let $\mathcal{I}$ be the set indexes



(a) Expected access delay. (b) Per-message power consumption. (c) Conditional success rate.

Figure 2.11: Performance of the proposed packet shedding scheme in Algorithm 1.

of the $|\mathcal{I}|$ previous packets delivered from α_j to a β-node. Denote $L_{\epsilon,i}(0)$ and $T_{CAD,i}$ as the initial distance and CAD of the i^{th} packet ($i \in \mathcal{I}$) generated by α_j, respectively. The packet shedding starts by calculating the average speed $\bar{v}$ and variance $\bar{\sigma}^2$ (see line 2 of Algorithm 1), which is used to estimate the successful access rate $\hat{\Pr}\{T_{CAD} < \tau\}$. To this end, we employ a worst-case estimation via the lower bound of $\hat{\Pr}\{T_{CAD} < \tau\}$ (see line 4). When $\hat{\Pr}\{T_{CAD} < \tau\}$ is able to exceed the required P_s, then the corresponding packet will be delivered to a β-node using SCF (see line 6). Otherwise, the packet is discarded to save energy.

After explaining the proposed packet shedding in Algorithm 1, we are ready to evaluate its performance.

Numerical evaluation

In every network realization, we generate a packet ϵ at an initial location that is selected at random within the cell area of $l \times l = (750m)^2$ and then apply the MoVe routing [99] with and without the proposed packet shedding scheme. The network parameters and simulation tools as outlined in Section 2.6.1 are re-employed. When packet shedding is used, we calculate the overall average speed $\bar{v}$ from the CAD of the previous $|\mathcal{I}| = 100$ delivered packets, while the packet deadline is set to $\tau = 5$ minutes, which is about the same order with the time until a responder team can reach an emergency location [88]. The target success rate is set to $P_s = 0.8$. All the presented results are averages over 3×10^4 network realizations.

First of all, we evaluate how the expected CAD will be impacted by the packet shedding, by plotting it versus the number of packet copies k in Fig. 2.11a. To this end, notice that the proposed packet shedding will discard packets that are predicted to have large CAD, e.g., those with $E(T_{CAD}) > \tau$. This is equivalent to applying lines 5-7 in Algorithm 1. As a result the expected delay of the remaining packets will effectively be decreased, compared to without

shedding, which is verified by comparing the red and blue bars in Fig. 2.11a.

Secondly, we evaluate how much transmit power can be conserved by employing the proposed packet shedding scheme. We start by focusing on the case without shedding (see the blue bars in Fig. 2.11b). Interestingly, the average per-packet transmit power consumption is almost constant with a value of $E(P_\epsilon) \approx 5.9$ W, irrespective of the number of copies k. The constant power is due to an existing trade-off: when k is small, the access delay is large; otherwise, the smaller delay nullifies the increase in k.

On the other hand, the packet shedding scheme discards packets that are predicted to have delay exceeding τ, thus equalizing the expected access delay across various k (see the red bars in Fig. 2.11a). Thus, the average power consumption $E(P_\epsilon)$ of the proposed packet shedding—which is equivalent to the (almost constant) average delay multiplied by the number of copies—becomes larger as k increases. In this case, the largest power consumption improvement of up to 24.35% is achieved with the smallest number of copies, that is, $k = 2$.

Finally, we evaluate the packet shedding's performance using a *conditional success rate*, which is defined as the number of deliverable packets over the number of non-discarded packets. To this end, Fig. 2.11c indicates that the conditional success rate corresponding to the case without shedding is an increasing function of k, due to the better opportunity of quickly reaching a cloud provided by the multi-paths. When packet shedding is employed, the conditional success rate is increased—with an average relative value of 8.69%—because the packets that tried to be delivered via SCF (i.e., satisfying line 5 of Algorithm 1) have a higher likelihood of reaching the β-node on-time.

2.7 Summary

We studied the problem of delivering a packet from a mobile IoT device to a cloudlet in disaster environments. To model the trajectory of the packet, caused by a store-carry-forward process, we introduced the notion of packet mobility that exhibits either supermartingale, martingale, or submartingale-like properties. We derived the lower and upper bounds of the cloudlet access delay (CAD), to show that it scales quadratically versus the initial packet-to-destination distance for supermartingale-like mobility. Otherwise, the scaling law increase is linear for martingale and submartingale-like mobilities. We further defined a successful access rate to quantify the likelihood that a time-critical application is supportable and analyzed its bounds. Using our analysis, we designed a scheme for identifying and shedding packets that are not deliverable within their time-critical delay constraint, which conserves up to 24.35% of transmission power. The work in this chapter provides not only a fundamental understanding regarding mobile-to-cloudlet delay, but also a practical method to quickly and efficiently deliver packets for IoT applications in hostile environments.

Chapter 3

Fast Rendezvous for Spectrum-Agile IoT Devices with Limited Channel Hopping Capability

3.1 Background and Motivation

Internet-of-Things (IoT) is an emerging paradigm in which smart objects (e.g., smartphones, tablets, wearables, and RFID tags) are equipped with wireless transceiver to make them an integral part of the Internet, in an attempt to seamlessly collect, efficiently process, and intuitively present information to the users [22]. It is expected that the number of IoT devices will reach an explosive 35 billion by 2012 [7], with an average of 5 devices around each person. Such a dense network of IoT devices will require wireless spectrum to be utilized efficiently. Unfortunately, existing wireless networks are regulated by a fixed spectrum policy that leads to the problem of inefficient spectrum usage [96]. To this end, *dynamic spectrum access* (DSA) [32, 173]—that enables IoT devices as unlicensed users (i.e., secondary users, SUs) to access unused licensed spectrum without interfering with primary users (i.e., primary users, PUs)—has been introduced to improve spectrum efficiency. By employing software-defined radio (SDR), IoT nodes can intelligently adjust their transmission parameters, such as transmit power, carrier frequency, and modulation scheme. As a result, the nodes can quickly switch their operation spectrum to utilize the unused spectrum efficiently.

In a DSA network, each IoT node is associated with a set of channels for communication, whose availability is determined by the behavior of the neighboring PUs. Depending on their locations, IoT nodes may have distinct sets of available channels because of the different neighboring PUs. Moreover, the neighboring PUs may change their transmission status, which result in the available channel sets that vary over time. To this end, the diverseness of available chan-

nel sets complicates the problem of establishing a common control channel, especially in a fully distributed network setting.

The most common way of establishing control channel among IoT nodes is by assigning one or more dedicated control channels for all nodes [173]. However, IoT nodes may be located at distinct positions and have different sets of available channels. As a result, the likelihood that every IoT nodes have one or more common control channels can be very low, especially when the network size is large. On the other hand, even though a global control channel is unoccupied and can be found, its availability may be varying over time. In particular, a PU may initiate a transmission over the control channel, and the lower priority IoT nodes must surrender the control channel. As the control channel becomes unavailable, IoT nodes cannot start any new data transmission, although there may still be other unoccupied common channels. The blocked SU transmission is exacerbated by the PUs that tend to occupy a channel for a long period once they started transmitting [133, 46], thus potentially causing a long channel blocking time. Finally, allowing multiple IoT nodes to use a few control channels may introduce bottleneck, especially in a high density environment.

To avoid the control channel bottleneck and long blocking time problems, channel hopping (CH) schemes are commonly used to find a common control channel in DSA networks [165, 45]. A CH scheme, programmed into every IoT node's radio, evenly divides the time into slots. Then the CH scheme generates the sequence of channels that the associated IoT nodes has to hop into at each time slot, which is referred as *CH pattern*, while guaranteeing that any pair of IoT nodes can *rendezvous* [165], i.e., hop to a common channel for exchanging control information such as the available channel information and link status. In the literature, there are two major ways for generating CH pattern: *random-based* [143, 49] and *sequence-based* schemes [53, 166, 46]. In the random-based schemes, each node randomly select an available channel according to a pre-determined distribution and then hop to the selected channel at each time slot, resulting in an expected time-to-rendezvous (TTR)—the delay from when a pair of IoT nodes start hopping until they hop into the same unoccupied channel—that scales with an order of $O(N^2)$, where N is the total number of available channels [143]. In the sequence-based schemes, on the other hand, each radio employs a fixed-length CH sequence that is repeated indefinitely. These CH schemes have been applied to networks with *heterogeneous* channels [159, 165], in which SUs have different spectrum sensing capabilities and different ranges of observable channels. Early sequence-based CH schemes had considered the *asymmetric* case, where one SU identified as a sender and another SU acting as a receiver respectively employ two different CH patterns (see ACH [31] and ARCH [47]). More recently, researchers have applied sequence-based CH schemes to the *symmetric* case in which the SUs employ identical hopping patterns irrespective of their roles (see SSCH [24], SYN-MAC [97], QCH [32], and DH-MAC[133]).

Despite the vast progress in the literature, designing a CH pattern for DSA networks with

IoT nodes remains a challenging problem, especially when the total spectrum is very wide. For example, the LTE-Unlicensed (LTE-U) standard—which opens the 5,150-5,850MHz unlicensed national information infrastructure (U-NII) bands for opportunistic access via DSA—provides up to 700MHz of total bandwidth [65]. Let the total bandwidth be divided into channels with uniform bandwidth B_0. In view of the vast spectrum bandwidth and channels, existing CH schemes simply assume that SUs are able to hop to any of the channels [159, 143, 49, 53, 166, 31, 47, 24, 97], which is unrealistic for IoT nodes due to the following factors. Firstly, every IoT node has an on-board SDR with an associated digital bandwidth of BW_{SDR} that limits the absolute frequency difference between the current and next-hop channels, given a pre-determined delay constraint [6]. For example, a state-of-art Ettus E310 USRP SDR can have up to $BW_{SDR} = 56MHz$ [11]. Switching to a new channel spaced BW_{SDR} apart from the current channel's frequency requires up to 3ms [6], while hopping to a channel outside this range takes more than 30ms [26]. Evidently, the latter is not sufficient to fulfill the CH hopping timeslot length of up to 10 millisecond, as required by existing LTE-U and IEEE 802.22 DSA standards [65]. Secondly, IoT nodes as the lower-priority users must not transmit on an occupied channel, while a wideband Nyquist sampling (WBNS) [142] for determining occupancy on *all* channels cannot be performed due to the limited BW_{SDR}. Instead, the nodes can only perform a partial-band Nyquist sampling (PBNS) [142] to a *subset* of channels, thus restricting the number of "safe" candidate channels for the next hop. In this paper, we assume that IoT nodes are restricted by the hopping range and sensing bandwidth limitations, and we refer such nodes to have *limited CH capability*.

Unfortunately, existing sequence-based CH schemes [31, 47, 24, 97, 32, 133] are not readily applicable to IoT nodes with limited CH capability. One of the main reasons is that these schemes generate a periodic CH sequence vector whose adjacent elements (i.e., channels at consecutive time slots) can be far-away from each other, thus violating the hopping range limit imposed by BW_{SDR}. As a result, existing sequence-based schemes must be rearchitected completely to account for the hopping range limit. Another reason is that sequence-based CH schemes generally require designing and storing a CH sequence vector with length that scales as $O(N^2)$ [165, 45]. Such a quadratic CH vector storage complexity is particularly imposing for future DSA networks like LTE-U, which will divide the U-NII bands into $B_0 = 100$kHz rasters that result in $N = 7,000$ total number of channels [65].[1] As a result, sequence-based CH schemes are not suitable for many IoT nodes, such as wearables and RFID tags, that are low-complexity devices with limited memory size.

Motivated by the lack of applicable schemes for IoT-based DSA networks, we design graph-

[1]Existing LTE-U standard divides the U-NII bands into 20Mhz channels, and reserves the 100kHz channel raster for future uses. We envision the smaller raster size will enable a fine-grained and efficient channel allocation for dense IoT-based DSA networks, which could be a norm in the future.

based CH schemes for rendezvous between IoT nodes. Specifically, we build *spectrum slice graphs* to capture the limited CH capability of IoT nodes, and then treat the proposed CH schemes as random walks over the graphs. Then, the random walks are configured so that the corresponding CH schemes achieve quick rendezvous irrespective of the choice of the initial channel hop, by minimizing a *maximum expected TTR* (MTTR) metric. To this end, our major contributions are outlined as follows.

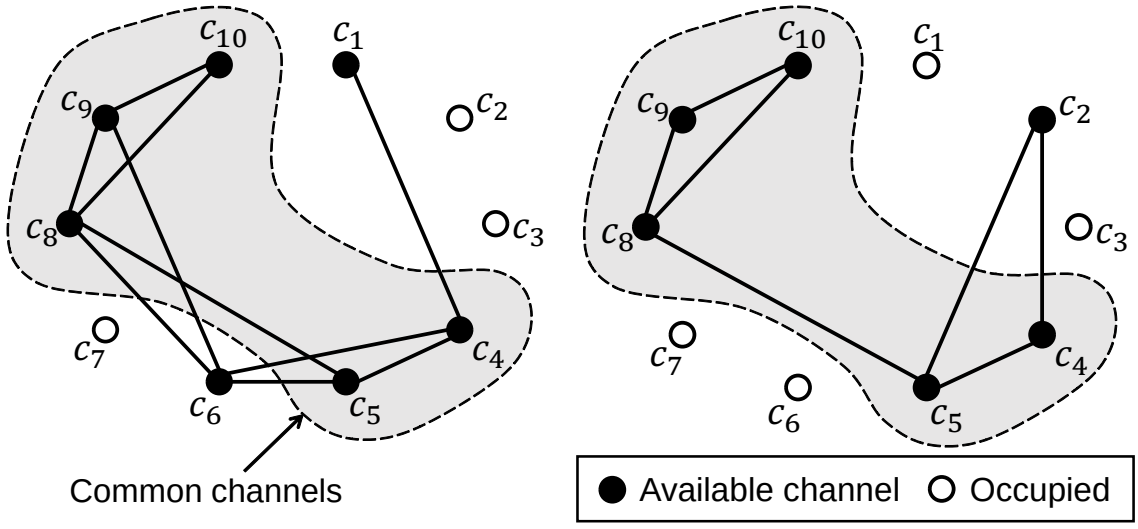
1. To the best of our knowledge, this study is the first to consider rendezvous between nodes with limited CH capability, which opens a new research avenue towards the design of low-complexity spectrum-agile devices.

2. We propose spectrum slice graphs to capture the possible CH sequences of a pair of IoT nodes, SU_a and SU_b, with limited CH capability. As exemplified by the spectrum slice graph of SU_a in Fig. 3.1a, the vertices (nodes) represent the set of available channels while their neighbors are the possible channels for the next hop. Moreover, we assume a *heterogeneous* model [159, 165] where the other node, SU_b, may have a different spectrum graph (see Fig. 3.1b).

3. We design a role-based random CH (R-RCH) scheme where node SU_a is set as a receiver that passively wait on a certain channel, while SU_b act as a sender that actively perform CH over its spectrum graph. By modeling SU_b's CH sequence as a random walk over the spectrum graph, we derive the upper bound of the corresponding MTTR, which is further minimized by assigning optimal transition probabilities to the walk. Then, we improve the performance of R-RCH by allowing the SU nodes to detect rendezvous on different but neighboring channels (vertices) of the spectrum slice graph.

4. We propose a symmetric random CH (S-RCH) scheme where both SU_a and SU_b actively walks over their respective spectrum slice graph. Using mathematical analysis, we derive the optimal transition probabilities for building the CH patterns of both IoT nodes. Unlike existing sequence-based schemes that require designing and storing long CH sequences [31, 47, 24, 97, 32], our random walk-based scheme provides a low-complexity, memoryless way of generating CH sequences on-the-fly.

5. We perform numerical evaluations and show that the proposed S-RCH scheme has a lower average TTR than the R-RCH scheme, which can outperform the state-of-art [143] when rendezvous is assisted by PBNS. We also show that the proposed schemes provide at least 80% successful rendezvous under realistic node mobility based on traces [130]. Finally, we employ experimental test bed based on USRP X310 SDRs to measure the energy consumption of the proposed scheme, and describe a practical scenario when R-RCH consumes less power and becomes more attractive than the S-RCH scheme.



(a) Graph $\mathcal{G}_a$ for node SU_a.

(b) Graph $\mathcal{G}_b$ for node SU_b.

Figure 3.1: Illustration of spectrum slice graphs.

The rest of this paper is organized as follows. Section 3.2 describes the network model and problem formulation. Sections 3.3 and 3.4 present the proposed R-RCH and S-RCH schemes, respectively, while their performance is analyzed in Section 3.5. Finally, Section 3.6 concludes this paper.

3.2 Models and Problem Formulation

In this section, we present the network model, channel hopping (CH) process for control channel establishment, and CH capability limits of practical IoT nodes. Then, we formulate the main problem of achieving quick rendezvous between nodes with limited CH capability.

3.2.1 Network Model

We consider a DSA network operating over an $(N \times B_0)$ Hz frequency spectrum that is divided into a set $\mathcal{C} = \{c_0, \ldots, c_{N-1}\}$ of non-overlapping channels, where c_i denotes the i-th channel, B_0 is the bandwidth of each channel, and N is the total number of channels. The DSA network consists of multiple PUs as the high-priority users and IoT nodes as the lower-priority SUs. Each SU is equipped with half-duplex SDR that can be tuned to any of the N channels. The SUs are also capable of performing spectrum sensing, to obtain their sets of available channels that are temporally unoccupied by co-located PUs. In view of the PUs' activity, the channel occupation status is assumed to be slowly-dynamic so that the DSA network is mildly time-varying.

We focus on the pairwise rendezvous problem, in which two IoT nodes—respectively denoted as SU_a and SU_b—want to establish a mutual control channel before data transmission. In essence, each of the IoT nodes try to rendezvous by scanning for unoccupied (available) channels, hop to one of these channels, and repeat these steps until a common control channel can be established. This process of finding a mutual control channel, referred as *channel hopping* (CH), is defined formally in the following subsection.

3.2.2 Channel Hopping for Control Channel Establishment

We consider a time-slotted DSA network where each slot—denoted as a non-negative integer t—has a uniform span of δ, which is the duration necessary for handshake and link establishment.[2] Let *available channels* $\mathcal{C}_a \subset \mathcal{C}$ and $\mathcal{C}_b \subset \mathcal{C}$ be the sets of unoccupied channels available to nodes SU_a and SU_b, respectively. The sets $\mathcal{C}_a$ and $\mathcal{C}_b$ are not necessarily identical to each other since SU_a and SU_b may be placed at different locations and surrounded by distinct sets of PU neighbors, which have independent channel occupation status. Although the sets of available channels are *heterogeneous* [159, 165], we assume that their intersection is non-empty (that is, $\mathcal{C}_a \cap \mathcal{C}_b \neq \emptyset$) to ensure that nodes SU_a and SU_b can still find a common channel for control channel establishment.

Let $X_a(t) \in \mathcal{C}_a$ be the channel that the IoT node SU_a hops into at time slot t, while the collection $\{X_a(t) : t \in N_0\}$ denotes the *CH sequence* of SU_a. Similar denotations exist for the other IoT node SU_b by exchanging $(\cdot)_a$ with subscript $(\cdot)_b$. Then, two SU nodes achieve *rendezvous* when they both hop to the same channel at the same time slot (i.e., $X_a(t) = X_b(t)$). Rendezvous enables a pair of IoT nodes to perform handshake and to exchange control information. The control information can further be used to initiate and regulate data transmission in the same channel (e.g., using in-band control) or a different frequency slice using out-of-band control.

3.2.3 Channel Hopping Involving Practical IoT Nodes

Future IoT-based DSA networks are envisioned to operate over a very wide frequency spectrum. For example, TV white space in the state of Kansas, USA provides DSA with up to 34 channels and 204 MHz of bandwidth [5], while LTE-U standard uses the U-NII spectrum that covers a total of 700MHz bandwidth [65]. In contrast to the massively available spectrum for DSA, however, realistic IoT nodes are restricted by their on-board SDR hardware that is designed to be portable, cheap to produce, and consume low power for maximizing battery lifetime. These practical considerations may prevent IoT nodes from leveraging the full available spectrum

[2]Each radio frame has a length of $\delta = 10ms$, according to LTE-U standard [65]. Similarly, IEEE 802.22 states that each frame has a length of $\delta = 10ms$.

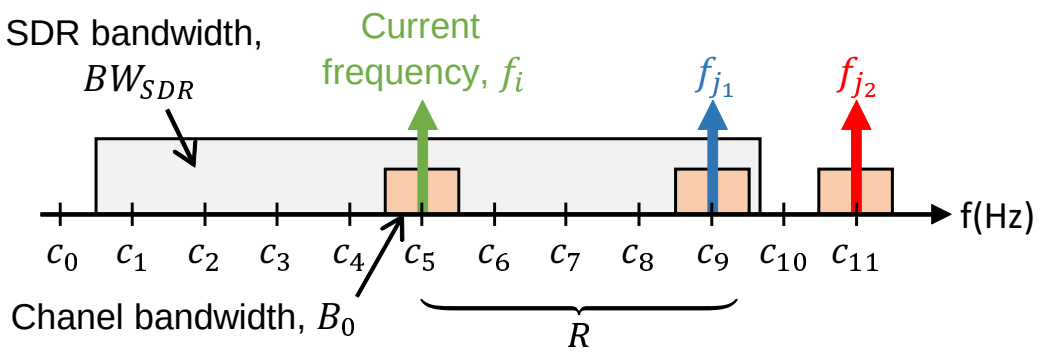


Figure 3.2: Illustration of frequency shifting limit.

during the CH process. As a result, IoT nodes have the following limitations.

Large frequency tuning time

Consider a CH process performed by SU_a (the following limitation also applies to SU_b's CH process). Let a channel $X_a(t) = c_i$ visited at time t has a center frequency of f_i Hz. To hop to c_i, the onboard SDR will tune its local oscillator's (LO's) frequency to f_i and then perform digital signal processing (DSP) to transmit and/or receive the corresponding signals for rendezvous. When SU_a wants to hop to a new channel $X_a(t+1) = c_j$ with a center frequency of f_j, the onboard SDR needs to perform one of the following tuning steps.

(a) The SDR has an associated digital bandwidth of BW_{SDR}[3] and it can perform a *one-stage tuning* [6] by shifting the signal's frequency digitally to a new channel within its digital bandwidth (e.g., to f_{j_1} in Fig. 3.2 that is located within $(\frac{BW_{SDR}}{2} - B_0)$ away from f_i). The latency for such a one-stage tuning is up to 3ms [11], which easily fits within a $\delta = 10ms$ time slot duration.[2]

(b) When SU_a decides to hop to a channel outside of the SDR's digital bandwidth (e.g., to f_{j_2} in Fig. 3.2), then the SDR needs to perform a *two-stage tuning* [6]. The SDR first sets the LO's frequency close to f_2 and then perform a digital frequency shift to account for the error in tuning, between the target f_{j_2} and the actual LO's frequency. This two-stage tuning involves re-setting and waiting for the LO's frequency to stabilize, which takes longer than the aforementioned one-stage tuning.

To understand the severity of the two-stage frequency tuning, we implement an experiment using a USRP X310 SDR equipped with an SBX-120 daughterboard. The USRP is connected

[3]For example, and X310 USRP SDR can have up to $BW_{SDR} = 160MHz$ if equipped with an UBX-160 daughterboard.

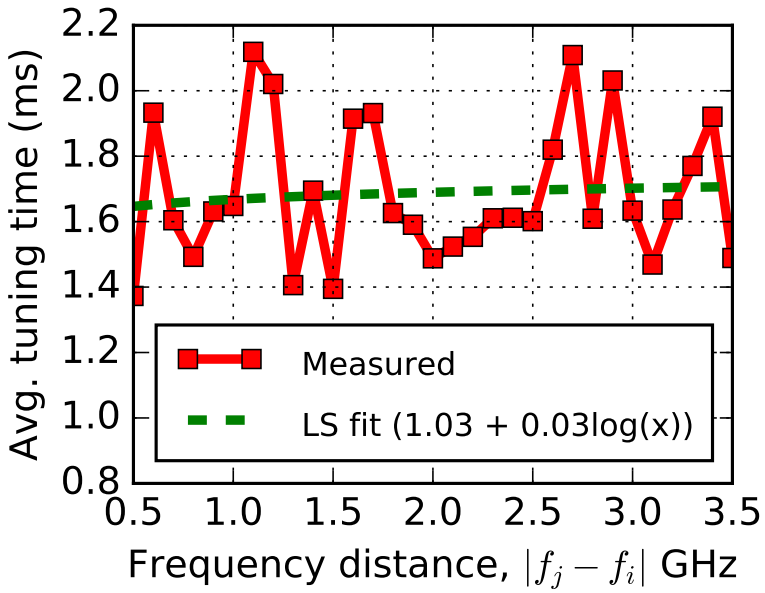


Figure 3.3: Average tuning time of USRP X310 test bed.

via 1000Base-T Ethernet to a Dell E5570 laptop running Ubuntu 16.04.2 LTS and a low-latency kernel on top of an Intel i7-6820HQ processor and a 16GB of DDR4 memory. The initial center frequency is set to $f_i = 500MHz$ and the two-stage tuning delay is measured as the time between a new center frequency f_j is requested until the `lo_locked` flag, which indicates the LO's frequency has stabilized, changes from 0 to 1. The plot of the average two-stage tuning delay over 10^3 experiments is depicted in Fig. 3.3.

Although there are irregularities in Fig. 3.3—possibly caused by the Linux operating system's scheduling algorithm and random internal states of the USRP when frequency change is requested, the overall trend indicates the tuning delay increases logarithmically versus the center frequency difference (e.g., $|f_i - f_j|$ Hz), verifying the existing result in [71]. From our measurements, the tuning delay is very small and lies somewhere between 1.2ms to 2.2ms. However, since the employed USRP X310 is one of the most advanced state-of-art SDR, the absolute tuning delay in Fig. 3.3 is lower than that of practical IoT nodes using generic SDR hardware. In fact, the tuning delay for generic SDRs typically reaches 48ms [11], which is way beyond the 10ms time slot period.[2] Thus, it is desirable to avoid the high-latency two-stage tuning.

To avoid two-stage tuning, we limit the frequency hopping range by keeping the distance

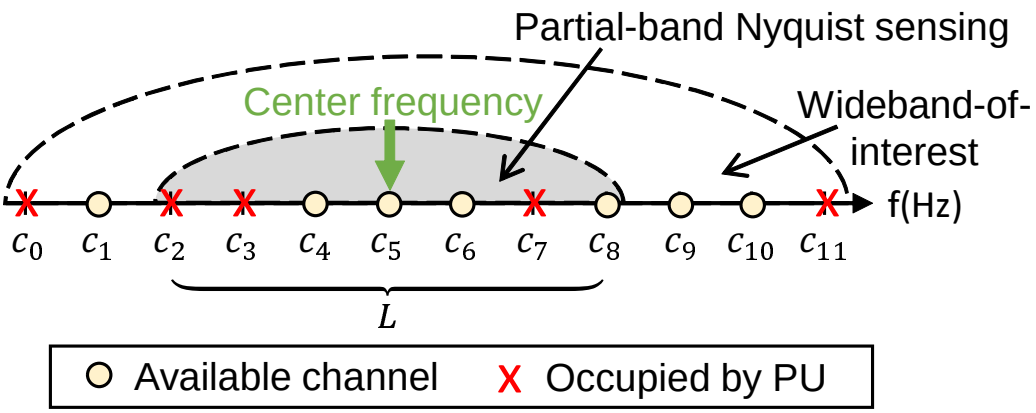


Figure 3.4: Illustration of channel sensing limit.

between channels at subsequent hops to be small, for ensuring a fast one-stage tuning. Let

$$R := \left\lfloor \frac{BW_{SDR} - B_0}{2B_0} \right\rfloor \tag{3.1}$$

be the *maximum CH range*. The numerator is subtracted with B_0 to ensure that the next hop's center frequency and the signal bandwidth B_0 can both fit into the SDR's bandwidth, BW_{SDR}. The factor 2 in the denominator is to ensure that the node can hop to the left and right sides of the current frequency f_i, with an equal distance (see Fig. 3.2). Then, an IoT node SU_a that currently visits channel $X_a(t) = c_i$ may hop to another channel $X_a(t+1) = c_j$ only if $|i-j| \leq R$. Note that similar limitation applies to the other node SU_b by replacing X_a with X_b, e.g., the channel hopped by SU_b.

Limited channel sensing capability

Before hopping to a new channel c_j, an IoT node (i.e., SU_a) must ensure that channel c_j is unoccupied to avoid interference to an ongoing PU transmission. In addition, the occupation status of all channels in the spectrum must also be checked to enable the identification of channels vacated by idle PUs. The vacated channels can then be added to the set of available channels, $\mathcal{C}_a$, to increase the possible number of common channels, $|\mathcal{C}_a \cap \mathcal{C}_b|$, and improve rendezvous opportunities.

Under the lack of spectrum database [5], especially when the DSA network is operating over an unlicensed band [65], the occupancy status must be checked via channel sensing. To this end, wideband Nyquist sampling (WBNS) [142] can be employed to sense the entire $(B \times N_0)$ Hz spectrum all at once, but imposes a high sampling rate requirement that is hard to be satisfied. For example, the LTE-U standard that covers a $B \times N_0 = 700MHz$ spectrum requires a Nyquist

sampling rate of 1.4 GHz, but state-of-art USRP SDR may only provide less that 200MHz of bandwidth.[3] On the other hand, sequential narrowband Nyquist sampling (SNNS) [142, 81]—that divides the spectrum into several *parts* (collection of channels), each with a bandwidth of BW_{SDR}, and then senses every part—involves multiple two-stage frequency tunings and requires a long time to complete, exceeding the time slot length of $\delta = 10ms$.[2] These restricted SDR bandwidth and tuning delay prevents WBNS and SNNS from being employed in practical DSA networks. Instead, real IoT nodes may only be able to perform a limited, one-time partial-band Nyquist sampling (PBNS) [142] that senses a part of the entire spectrum and ignores the remaining parts.

To model the impact of PBNS-induced sensing capability limit, we consider a CH process at node SU_a (a similar model also holds for SU_b). When SU_a hops to the current channel $X_a(t) = c_i$, it will set the on-board SDR's LO frequency to f_i, which also becomes the PBNS's center frequency. Let

$$L := \left\lfloor \frac{BW_{SDR}}{2B_0} \right\rfloor \tag{3.2}$$

be the number of channels that can be sensed by PBNS. The factor 2 in the denominator is because according to Nyquist theorem, a perfect reconstruction of a signal with bandwidth B_0 needs to be sampled at a rate of at least $2B_0$. Then, PBNS is capable of sensing channels c_x to c_y, where $x := \max(0, i - \lfloor \frac{L-1}{2} \rfloor)$ and $y := \min(i + \lfloor \frac{L-1}{2} \rfloor, N-1)$ are the lower and upper bounds of the sensible channel indexes. For example, a PBNS scheme with a center frequency of $f_i = f_5$ and $L = 7$ in Fig. 3.4 can sense channels $c_x = c_2$ to $c_y = c_8$, and then determines that $\{c_4, c_5, c_8\}$ are unoccupied. These unoccupied channels then serve as the candidates for the next hop.

3.2.4 Rendezvous for IoT Nodes With Limited CH Capability

In practical DSA networks, the frequency tuning delay and channel sensing capability limits not only co-exist, but also impose a *combined CH range* of

$$R_{ch} := \min\left(R, \left\lfloor \frac{L-1}{2} \right\rfloor\right). \tag{3.3}$$

To be specific, an IoT node that currently visits channel c_i may hop to another channel c_j if and only if the CH range restriction $|i - j| \leq R_{ch}$ is satisfied. In this paper, all IoT nodes are assumed to have identical CH ranges, and we refer the nodes with such a restriction to have a *limited CH capability*. The limited CH capability will reduce the set of available channels for the next hop, which is a key ingredient that determines when and where rendezvous between SU_a and SU_b occurs. To describe the reduced set of next-hop channels, we define the following.

Definition 9 (Spectrum slice graph)**.** *Consider an IoT node SU_x, where $x \in \{a, b\}$ denotes an index, and a maximum channel hopping range of R_{ch}. Let vertices $\mathcal{C}_x$ represent the set of channels available to node SU_x. Denote $\mathcal{E}_x$ as the set of edges, where an edge $e(c_i, c_j)$ is in $\mathcal{E}_x$ if $c_i, c_j \in \mathcal{C}_x$ and $|i - j| \leq R_{ch}$. Then, the spectrum slice graph of node SU_x is defined as a tuple $\mathcal{G}_x := (\mathcal{C}_x, \mathcal{E}_x)$.*

An example of a spectrum slice graph for node SU_a that corresponds to the channel sensing limit in Fig. 3.4 is depicted in Fig. 3.1a. The vertices in Fig. 3.1a are the set of available channels while edges represent the connections to the set of available channels for the next hop, which is restricted according to (3.3). For example, SU_a that is currently at channel (vertex) c_5 may only choose the neighbors c_4, c_5, c_6, or c_8 for the next time slot.[4] This is because SU_a is unable to sense the occupancy of other channels outside this range (i.e., $c_9 - c_{11}$). Attempting a rendezvous on these outside channels may cause strong interference to co-located PUs and should be avoided.

Recall that according to Section 3.2.2, we assume a heterogeneous available channel assumption. As a result, the other node, SU_b, may have a spectrum slice graph $\mathcal{G}_b$ that is different than SU_a's graph (see Fig. 3.1b).

The spectrum slice graphs are then used by their respective SU nodes to determine the sequence of channels they need to visit at each time slot, which is rigorously defined below.

Definition 10 (CH sequence under limited capability)**.** *Let $c_{i_t} := X_x^{ch}(t) \in \mathcal{C}_x$ be the channel visited by node SU_x ($x \in \{a, b\}$) at time t and $\mathcal{N}_x(c_{i_t}) := \{c : e(c_{i_t}, c) \in \mathcal{E}_x\}$ be the neighbors of vertex c_t in the spectrum slice graph $\mathcal{G}_x$. Given c_{i_t} and the CH capability limit of SU_x, the next-hop channel $c_{i_{t+1}} := X_x^{ch}(t+1)$ must be selected among $\mathcal{N}_x(c_{i_t}) \cup \{c_{i_t}\}$, that is, the neighbors of and c_{i_t} itself. Then, the CH sequence of SU_x is defined as a collection*

$$W_x := \{X_x^{ch}(t) : t \in N_0\}. \tag{3.4}$$

An illustration of a CH sequence corresponding to node SU_b that starts by visiting channel $X_a^{ch}(0) = c_8$ is depicted in Fig. 3.5. A similar CH sequence also exists for node SU_a but is omitted for conciseness. In Fig. 3.5, channel (vertex) c_8 is connected via edges to vertices $\mathcal{N}_b(c_8) = \{c_5, c_9, c_{10}\}$ and the next-hop channel must be selected among $\mathcal{N}_b(c_8) \cup \{c_8\}$. In this example, SU_b decides to hop to $X_b^{ch}(1) = c_9$ at the next time slot, $t = 1$. By repeating the aforementioned process, we obtain a CH sequence of $W_b = \{c_8, c_9, c_{10}, c_8, \ldots\}$, as depicted by the red arrows in Fig. 3.5. To this end, the path formed by the CH sequence in Fig. 3.5 is essentially a walk on graph $\mathcal{G}_b$. We assume that the next-hop channel is selected randomly according to

[4]Without loss of generality, we allow the node to re-choose the existing channel c_5 for the next hop.

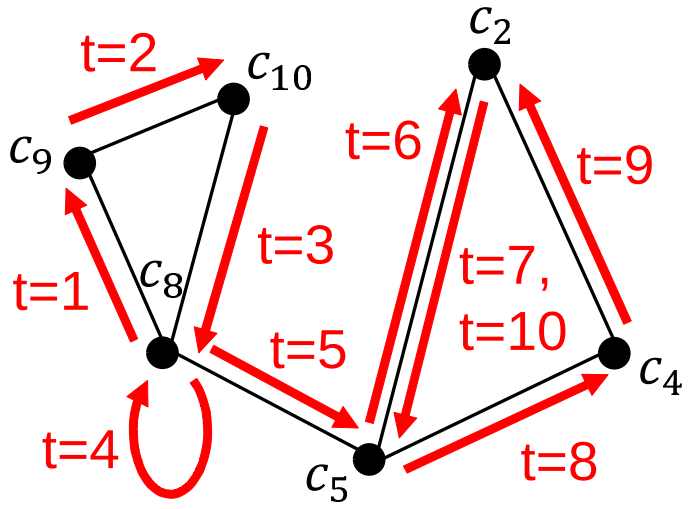


Figure 3.5: CH sequence of SU_b illustrated as a walk over the spectrum slice graph $\mathcal{G}_b$ in Fig. 3.1b.

a pre-determined probability distribution, such that the CH sequence becomes a *random walk* [100] over a spectrum slice graph. The problem of assigning the optimal probability distribution, on the other hand, will be detailed in Sections 3.3.3 and 3.4.2 below.

Based on the CH sequence in Definition 10, the rendezvous problem between nodes with CH capability limits can be formulated as follows.

Definition 11 (Rendezvous)**.** *A rendezvous between nodes SU_a and SU_b is achieved if there exists $c^* \in \mathcal{C}_a \cap \mathcal{C}_b$ and $t^* \in N_0$ such that $X_a^{ch}(t) = X_b^{ch}(t) = c^*$.*

In other words, rendezvous occurs when the (random) walks performed by SU_a and SU_b over their respective spectrum slice graphs both visit the same channel (vertex).

3.2.5 Problem Formulation

The primary goal behind the rendezvous problem is to quickly establish a common control channel between two IoT nodes in a distributive manner. Therefore, delay performance is of critical importance, and we focus on minimizing the delay until rendezvous can be achieved. To achieve our minimization goal, we employ the notion of *time-to-rendezvous*

$$TTR := \min\{t \geq 0 : X_a^{ch}(t) = X_b^{ch}(t) = c^*, c^* = \mathcal{C}_a \cap \mathcal{C}_b\} \tag{3.5}$$

from [165], and then use TTR to define the following metric.

Definition 12. *Maximum expected time-to-rendezvous (MTTR) is the largest average TTR among all possible initial channel hops,* $X_a^{ch}(0)$ *and* $X_b^{ch}(0)$*, and is defined as*

$$TTR_{max} := \sup_{X_a^{ch}(0), X_b^{ch}(0)} E(TTR). \tag{3.6}$$

When MTTR is minimized, nodes can establish a control channel and start data transmission immediately, which leads to quick medium access delay and high user satisfaction. The problem of designing CH sequences for quick MTTR in idealized DSA networks without CH capability limit has been studied extensively [143, 49, 31, 47, 24, 97, 32, 133]. However, minimizing MTTR between two IoT nodes with limited CH capability remains a wide-open problem. Hence, we ask: *How to design the CH sequences of IoT nodes with limited CH capability for achieving a fast MTTR?*

3.3 Role-Based Random Channel Hopping (R-RCH)

According to the literature, an IoT-based network comprises two types of nodes: (i) *distributed sensors and actuators,* which harvest data from the environment, and (ii) a *point-of-access device* (AP) that collects data from the distributed nodes [78]. Several IoT examples of such a master (AP)-slave (sensor/actuator) architecture includes the collection of consumer-side electrical energy usage via advanced metering infrastructure in smart grid [148], and air quality monitoring via geographically-distributed sensors in smart cities [171]. In these scenarios, the DSA network needs to facilitate the establishment of control channel between an AP (master) and a distributed sensor/actuator (slave) node via rendezvous. To this end, a rendezvous scheme can leverage the distinct properties of these two nodes by assigning them with two different roles.

Role-based rendezvous, also known as *asymmetric* method [165], has long been achieved by letting the IoT nodes to apply periodic CH sequences [31, 47]. Unfortunately, the existing asymmetric schemes are based on the assumption that IoT nodes can hop to any arbitrary channel, neglecting the CH capability limit of practical SDRs. Moreover, these schemes require the design and storage of CH sequence with length $O(N^2)$, which is prohibitive in future DSA networks that may have up to $N = 7000$ channels.[1] Asymmetric CH scheme based on random sequences, on the other hand, provides a low-complexity alternative without the hassle of a long CH sequence [143]. However, like their sequence-based counterparts [31, 47], existing role-based random CH schemes are unsuitable and need to be re-designed to account for IoT nodes' CH capability limit. Motivated by the lack of solution that accounts for the nodes' CH capability limit, we design a role-based random channel hopping (R-RCH) scheme that also achieves our

Algorithm 2 Child-role channel hopping

Set the initial time: $t \leftarrow 0$.
Apply PBNS to select an unoccupied channel $X_a(0) = c_i$.
Hop to c_i and attempt to rendezvous.
while rendezvous not achieved **do**
    Increment the time: $t \leftarrow t + 1$.
    if ($t \mod M_s == 0$) **then**
        Perform PBNS centered at f_i to find $\mathcal{N}_a(c_i)$.
        Select c_j uniformly at random from $\mathcal{N}_a(c_i) \cup \{c_i\}$.
        Hop to c_j by assigning $X_a(t) = c_i \leftarrow c_j$.
    Attempt to rendezvous on c_i.

main objective of minimizing the MTTR.

3.3.1 Preliminaries on R-RCH Scheme

R-RCH is inspired by the "wait-for-mommy" (WFM) scheme from wildlife [20]. In WFM, a child node stays in its location when being lost, while a mommy node exhaustively explores all possible locations, thus guaranteeing rendezvous. We adopt the bio-inspired WFM approach by considering a sensor/actuator as SU_a and an AP as SU_b, then assign each node with different role. This is a valid assumption since most control channel establishment requests are between the sensor/actuator and AP nodes, for enabling the information harvested by the former to be collected by the latter. Inter-sensor/actuator control channel establishment, on the other hand, is unnecessary since data exchange among these nodes can be facilitated by the AP. To focus on the rendezvous between a sensor/actuator (SU_a) and an AP (SU_b), we assign them with the following policies and call the resulting scheme as R-RCH.

Child-role CH process for SU_a

In the context of IoT network, SU_a is a sensor/actuator node that have low hardware complexity and limited energy source, thus preferring a simple operation. Thus, we assign node SU_a with a simpler role, similar to a child in WFM, which is outlined as follows.

Node SU_a starts by selecting one of the vertices in its spectrum slice graph $\mathcal{G}_a$, to hop to an available channel. Technically, this can be done by initially applying PBNS sensing centered at an arbitrary frequency and then selecting an unoccupied channel $X_a^{ch}(0) = c_i$ at the first hop. Conventional WFM algorithm [143] requires SU_a to stay and wait indefinitely for its counterpart, node SU_b, in channel c_i. The drawback of the conventional approach is SU_a may wait on a channel (vertex) that does not exist in SU_b's spectrum graph, $\mathcal{G}_b$. For example, SU_a with a spectrum slice graph in Fig. 3.1a may select c_1 that is not in the spectrum slice graph of

Algorithm 3 Mommy-role channel hopping

1: Set the initial time: $t \leftarrow 0$.
2: Apply PBNS, select an unoccupied channel $X_b(0) = c_i$.
3: Hop to c_i and attempt to rendezvous.
4: **while** rendezvous not achieved **do**
5: Increment the time: $t \leftarrow t + 1$.
6: Perform PBNS centered at f_i to find $\mathcal{N}_b(c_i)$.
7: Select c_j from $\mathcal{N}_b(c_i)$ according to $P(c_i, c_j)$.
8: Hop to c_j by assigning $X_b(t) = c_i \leftarrow c_j$ and attempt rendezvous.

SU_b (see Fig. 3.1b). As a result, rendezvous will never occur since SU_b cannot hop into channel c_1.

To alleviate the aforementioned non-rendezvous issue, we allow the "child" node SU_a to hop to the current channel's neighbor every $M_s > 0$ time slots.[5] To this end, the resulting CH sequence can be viewed as a walk over $\mathcal{G}_a$ where a new step happens every M_s time slots, as outlined in Algorithm 2.

Mommy-role CH process for SU_b

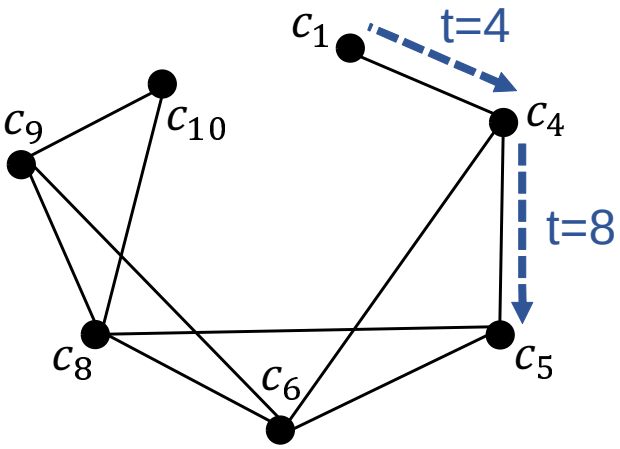
Node SU_b, on the other hand, represents an AP device that typically has a more advanced hardware specification. Thus, SU_b is able to implement a more active and complex role as a mommy.

As a mommy, SU_b initially hops to one of its available channels by randomly selecting a vertex in graph $\mathcal{G}_b$. Then, SU_b searches for the (relatively) static child node SU_a by applying the CH scheme in Definition 10. Specifically, SU_b selects the next-hop channel from either the current-hop channel c_i or its neighbors, $\mathcal{N}_b(c_i)$, as outlined in Definition 10. The question is, how to select the next channel that achieves our main objective of minimizing the MTTR? To answer the question, let *transition probabilities* of SU_b be denoted as $\{P_b(c_i, c_j) : c_i, c_j \in \mathcal{C}_b\}$, where

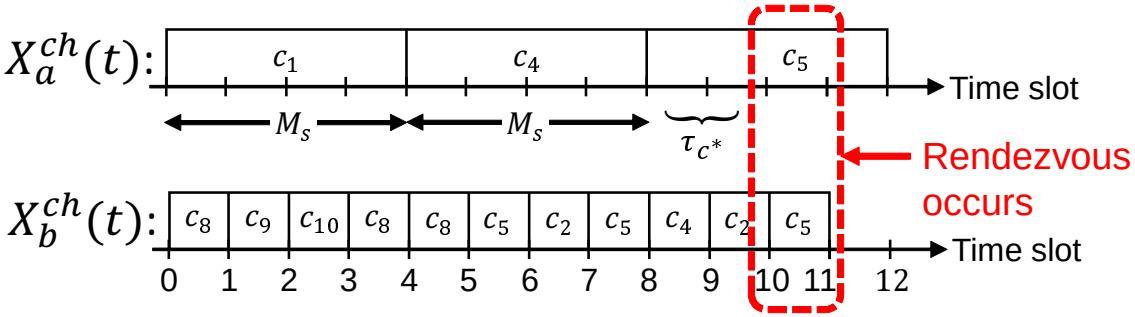
$$P_b(c_i, c_j) := \Pr\{X_b(t+1) = c_j | X_b(t) = c_i\} \tag{3.7}$$

is the likelihood that SU_b will select a next-hop channel c_j given an existing channel c_i. Together, the CH sequence and transition probabilities form a random walk over the spectrum slice graph $\mathcal{G}_b$ (see Fig. 3.5 for an example of such a walk). It is known that a random walk is fully determined by its transition probabilities [100]. Hence, the performance of SU_b's CH sequence, which is essentially a random walk over graph $\mathcal{G}_b$, can be optimized by carefully assigning

[5] In practice, the neighbors of the current channel c_i can be determined on-the-fly by applying a PBNS centered at c_i prior to the next hop.



(a) Channel hopping (CH) sequence of SU_a over the spectrum slice graph $\mathcal{G}_a$ in Fig. 3.1a.



(b) CH sequences of SU_a and SU_b until rendezvous.

Figure 3.6: Illustration of role-based random channel hopping (R-RCH) scheme.

the transition probabilities. In Section 3.3.3, we show how to derive the optimal transition probabilities so that SU_b's CH sequence results in a minimized MTTR.

3.3.2 An Illustration of R-RCH

To better understand how R-RCH works, we consider an example in which the child node SU_a hops to a new channel every $M_s = 4$ time slots, as depicted in Fig. 3.6a. When SU_a does not hop to a new channel (e.g., at times $t \in (5, 7)$ and $t \in (8, 11)$), it attempts rendezvous on the current channel. The resulting CH sequence, $W_a = \{X_a^{ch}(t)\}$, is depicted in the top part of Fig. 3.6b. On the other hand, the counterpart node, SU_b, applies the mommy-role CH process in Algorithm 3 that produces a random walk as depicted in Fig. 3.5. The resulting CH sequence, $W_b = \{X_b^{ch}(t)\}$, is illustrated in the bottom part of Fig. 3.6b. Notice how SU_b as the mommy hops at every time slot, faster than that of the child node, SU_a. By examining the CH sequences

in Fig. 3.6b, we find that SU_a and SU_b visits a common channel c_5 and achieve rendezvous at time $t = 10$.

3.3.3 R-RCH with Quick Rendezvous

We have outlined how an R-RCH scheme consists of a relatively static child node SU_a and a mommy node SU_b that actively searches the child by performing random walk. In this subsection, we discuss how to design the transition probabilities of SU_b's random walk for minimizing MTTR. According to Definition 12, our design objective will ensure a quick rendezvous delay, regardless of the initial channels (vertices) selected by the child and mommy nodes.

To proceed, let the *degree* of vertex (channel) c_i in the spectrum slice graph $\mathcal{G}_b$, denoted as $d_b(c_i) := |\mathcal{N}_b(c_i)|$, be the number of edges connected to c_i. For example, there are three edges connected to vertex c_8 in Fig. 3.6a. Then, we have the following theorem.

Theorem 4. *The transition probabilities that achieve the smallest MTTR satisfy,* $\forall c_i, c_j \in \mathcal{C}_b$,

$$P_b(c_i, c_j) \;=\; \begin{cases} \frac{1}{d_b(c_i)+1}, & \text{if } e(c_i, c_j) \in \mathcal{E}_b \text{ or } c_i = c_j, \\ 0, & \text{otherwise.} \end{cases} \tag{3.8}$$

Before proving Theorem 4, we introduce a few notations and lemmas. Let $n_b := |\mathcal{C}_b|$ and

$$\mathbf{P}_b := \{P_b(c_i, c_j)\}_{c_i, c_j \in \mathcal{C}_b} \tag{3.9}$$

be the $n_b| \times n_b|$ *transition probability matrix* that satisfies the *stochastic* property, i.e., every row sums to one. Denote $P_b^t(c_i, c_j)$ as the *conditional* likelihood that a random walk currently located at c_i becomes located at c_j after t time slots, which is the (i, j)-th element of $\mathbf{P}_b^t = \mathbf{P}_b^{t-1}\mathbf{P}_b$ with $\mathbf{P}_b^1 = \mathbf{P}_b$. Moreover, let

$$\boldsymbol{\pi} := \{\pi_{c_i}\}_{c_i \in \mathcal{C}_b} \tag{3.10}$$

be a *stationary distribution* vector that satisfies $\boldsymbol{\pi}\mathbf{P}_b = \boldsymbol{\pi}$ and $\boldsymbol{\pi}\mathbf{1} = \mathbf{1}$, where $\pi_{c_i} \in (0, 1)$ is the *unconditional* likelihood that SU_b will be at channel c_i when it has hopped a sufficiently large number of steps. Then, we have the following definition.

Definition 13. *[100] Given* $\epsilon > 0$*, the mixing time of a random walk with a transition probability matrix* $\mathbf{P}_b$ *is defined as*

$$\tau_{mix}(\epsilon) := \inf\left\{t : \max_{c_i \in \mathcal{C}_b} \left|P_b^t(c_i, \cdot) - \pi_{(\cdot)}\right| \le \epsilon\right\}. \tag{3.11}$$

The mixing time is the amount of steps necessary until the transition probability of the walk over graph $\mathcal{G}_b$ is close enough (with a factor ϵ) to its stationary distribution. The mixing time will play a key role in the derivation of Theorem 4.

Besides the mixing time, another essential ingredient towards proving Theorem 4 is to derive the MTTR for R-RCH. However, analyzing the exact expression of MTTR is hard so that we derive the following upper bound, instead. Denote $q_a := \frac{|\mathcal{C}_a \cap \mathcal{C}_b|}{|\mathcal{C}_a|}$ as the ratio between the number of common channels to the number of available channels at the receiver SU_a. Moreover, let $\tau_L(\epsilon)$ be the mixing time of a *lazy walk*, that is, a random walk over $\mathcal{G}_b$ with transition probability $\mathbf{P}_L = \frac{1}{2}(\mathbf{P}_b + \mathbf{I})$. Then, we have the following lemma.

Lemma 3. *Let c'_α be a positive constant. Then, the MTTR of the R-RCH scheme is upper-bounded by*

$$TTR_{max} \leq \left(\frac{1-q_a}{q_a}\right) \times M_s + \frac{1}{c'_\alpha}\tau_L\left(1/4\right). \tag{3.12}$$

Proof. According to Definition 11, rendezvous may only occur if the child node SU_a and the mommy node SU_b hops to one of the common channels, $\mathcal{C}_a \cap \mathcal{C}_b$. To leverage this necessary condition and determine the TTR, we first focus on the child node SU_a, which hops to one of its available channels uniformly at random every M_s time slots and results in a probability q_a of selecting a common channel. As a result, the number of attempts until SU_a selects a common channel, denoted as k, is Binomial-distributed with probability density function $P(k) = (1-q_a)^{k-1}q_a$. According to the above-mentioned CH selection process, node SU_a will hop to a common channel $c^* \in \mathcal{C}_a \cap \mathcal{C}_b$ at time $t' = (k-1)M_s + 1$. We assume that the time until SU_a hops to a new channel, M_s, is sufficiently large such that there is enough time to ensure SU_a and SU_b can rendezvous in the selected common channel.

To proceed, we denote $\tau_{c^*} := \min\{t - t' : t \geq 0, X_b(t) = c^*\}$ as the time starting at when SU_a hops to a common channel c^* until it can rendezvous with the mommy node SU_b. Then, the TTR can be re-stated as

$$TTR = (k-1)M_s + \tau_{c^*}. \tag{3.13}$$

By applying a maximization and expectation operator to (3.13), we can obtain the following inequality.

$$\begin{aligned} \max_{c^*, c_i \in \mathcal{C}_b} E[TTR] &= \max_{c^*, c_i \in \mathcal{C}_b} \{M_s \times (E[K]-1) + E_{c_i}[\tau_{c^*}]\} \\ &\leq M_s\left(\frac{1}{q_a} - 1\right) + \frac{1}{c'_\alpha}\tau_L(1/4). \end{aligned} \tag{3.14}$$

To prove the last inequality, let

$$t_H(\alpha) := \max_{c_i, C:\pi(C)>\alpha} E_{c_i}[\tau_C], \tag{3.15}$$

for $\alpha < \frac{1}{2}$, be the expected hitting time of a large set $C \in \mathcal{C}_b$ [117]. Then, the inequality holds since $\max_{c^*, c_i \in \mathcal{C}_b} E_{c_i}[\tau_{c^*}] \leq t_H(\alpha)$ and $t_H(\alpha)$ is upper-bounded by $\frac{1}{\mathcal{C}'_\alpha}\tau_L(1/4)$ [117, Thm. 1.1]. Finally, Lemma 3 can be obtained by plugging the the definition of MTTR in (3.6) to left hand-side of (3.14). □

Lemma 3 states that the upper bound of MTTR in (3.6) is proportional to the mixing time of a lazy walk, $\tau_L(1/4)$. Notice that $\mathbf{P}_L^t = (\mathbf{P}_b + \mathbf{I})^t$ and the norm $|\frac{1}{2}(\mathbf{P}_b + \mathbf{I})^t(c_i, \cdot) - \pi_{(\cdot)}|$ in $\tau_L(1/4)$ are increasing functions of the transition probability matrix $\mathbf{P}_b$. Moreover, $\tau_L(1.4)$ is an increasing function of $\tau_{mix}(\epsilon)$, since $\tau_{mix}(\epsilon)$ in (3.32) is also monotonically-growing with respect to $\mathbf{P}_b$. As a result, achieving a fast MTTR via reducing $\tau_L(1.4)$ in (3.12) is equivalent to minimizing $\tau_{mix}(\epsilon)$.

To leverage the proportionality between MTTR and the mixing time $\tau_{mix}(\epsilon)$ for designing the transition probabilities $\{P_b(c_i, c_j)\}$, we further study the relation between $\tau_{mix}(\epsilon)$ and the eigenvalues of the transition probability matrix $\mathbf{P}_b$. Let λ be an *eigenvalue* of matrix $\mathbf{P}_b$ if there exists a non-zero vector $\mathbf{v}$ that satisfies $\mathbf{P}_b\mathbf{v} = \lambda\mathbf{v}$. In general, the transition probability matrix has n_b eigenvalues denoted as $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{n_b}$, where λ_i is the i^{th} largest eigenvalue. Let

$$\lambda^* := \max\{|\lambda_2|, |\lambda_{n_b}|\} \tag{3.16}$$

be the *second largest eigenvalue magnitude* (SLEM) of $\mathbf{P}_b$. Then, we have the following lemma.

Lemma 4. *[100] Let $\pi_{min} := \min_{c_i \in \mathcal{C}_b} \pi_{c_i}$. The mixing time of a random walk with transition probability matrix $\mathbf{P}_b$ is bounded by*

$$\begin{aligned}\left(\frac{1}{1-\lambda^*} - 1\right)\log\left(\frac{1}{2\epsilon}\right) &\leq \tau_{mix}(\epsilon) \\ &\leq \left(\frac{1}{1-\lambda^*}\right)\log\left(\frac{1}{\epsilon\pi_{min}}\right).\end{aligned} \tag{3.17}$$

The lower and upper bounds of $\tau_{mix}(\epsilon)$ in (3.17) contain a common term $\frac{1}{1-\lambda^*}$, where $(1-\lambda^*)$ is called the *spectral gap* [34], such that finding the smallest SLEM λ^* is equivalent to minimizing $\tau_{mix}(\epsilon)$. On the other hand, Lemma 3 states that minimizing $\tau_{mix}(\epsilon)$ translates to minimizing MTTR. As a result, Lemma 3 and 4 jointly indicates that our main objective of reducing MTTR can be achieved by minimizing λ^*. In the following, we leverage such a finding to prove Theorem 4.

Proof of Theorem 4. Recall that $\mathbf{P}_b$ is a stochastic matrix. Thus, $\mathbf{P}_b$ has a largest eigenvalue of $\lambda_1 = 1$ [100, Sec. 12.2], implying that the other eigenvalues are smaller or equal to one, so that the SLEM becomes $\lambda^* \leq 1$. Let $\mathbf{I}$ be an identity matrix. When $\mathbf{P}_b$ is projected to the null space of $\mathbf{1}$, that is, by using a projection function

$$(\mathbf{I} - \frac{1}{n_b}\mathbf{1}\mathbf{1}^T)\mathbf{P}_b(\mathbf{I} - \frac{1}{n_b}\mathbf{1}\mathbf{1}^T), \tag{3.18}$$

the largest eigenvalue magnitude is equal to the SLEM [35]. Moreover, since the *spectral norm* $||\mathbf{A}||_2 := \sup_{||\mathbf{x}||\geq 0} \frac{||\mathbf{A}\mathbf{x}||_2}{||\mathbf{x}||_2}$ is equal to the largest eigenvalue magnitude of $\mathbf{A}$, then the projection $||(\mathbf{I} - \frac{1}{n_b}\mathbf{1}\mathbf{1}^T)\mathbf{P}_b(\mathbf{I} - \frac{1}{n_b}\mathbf{1}\mathbf{1}^T)||_2 = ||\mathbf{P}_b - \frac{1}{n_b}\mathbf{1}\mathbf{1}^T||_2$ becomes equal to the SLEM λ^*. As a result, the problem of minimizing the mixing time $\tau_{mix}(\epsilon)$, which is equivalent to minimizing λ^*, can be re-stated as

$$\begin{aligned} \min_{\mathbf{P}_b} \left|\left| \mathbf{P}_b - \frac{1}{n_b}\mathbf{1}\mathbf{1}^T \right|\right|_2 \text{ s.t. } \mathbf{P}_b\mathbf{1} = \mathbf{1},\ \mathbf{P}_b \geq \mathbf{0}, \\ \text{and } P_b(c_i, c_j) = 0, \forall c_i, c_j : e(c_i, c_j) \notin \mathcal{E}_b \text{ and } c_i \neq c_j. \end{aligned} \tag{3.19}$$

The inequality $\mathbf{P}_b \geq \mathbf{0}$ indicates that the transition probability matrix has non-negative elements, while the last constraint ensures that the random walk is only performed over the edges of the spectrum slice graph $\mathcal{G}_b$.

Solving (3.19) requires the full knowledge regarding the set of edges $\mathcal{E}_b$ for assigning the elements of $\mathbf{P}_b$ all at once [35], which is not available to node SU_b due to the limited sensing capability of the employed PBNS algorithm. When SU_b is at channel c_i, however, it knows the immediate set of edges connected to c_i, which can be used for determining the i^{th} row of $\mathbf{P}_b$. Let $\mathbf{p}(c_i)$ be the i^{th} row of matrix $\mathbf{P}_b$. Further, we notice that the spectral norm is upper-bounded by the *Frobenius norm* $||\mathbf{A}||_F := (\sum_{i,j} A_{i,j}^2)^{\frac{1}{2}}$, where $A_{i,j}$ denotes the (i,j)-th element of matrix $\mathbf{A}$. As a result, (3.19) can be restated into the problem of assigning the u^{th} row of $\mathbf{P}_b$ as follows. Let $p_j := P_b(c_i, c_j)$ and $\mathbf{p}_b(c_i) := \{p_j\}_{c_j \in \mathcal{C}_b}$ for notation simplicity.

$$\begin{aligned} \min \left|\left| \mathbf{p}_b(c_i) - \frac{1}{n_b}\mathbf{1} \right|\right|_F \text{ s.t. } \mathbf{p}_b(c_i)\mathbf{1} = 1, \\ \text{and } P_b(c_i, c_j) \begin{cases} = 0, & \text{if } e(c_i, c_j) \notin \mathcal{E}_b \text{ and } c_i \neq c_j. \\ \geq 0, & \text{otherwise,} \end{cases} \end{aligned} \tag{3.20}$$

The optimal solution to (3.20) can be found as follows. Due to the constraint $\forall c_j \in \mathcal{C}_b \setminus \mathcal{N}_b(c_i), p_j = 0$, the Lagrangian of (3.20) becomes $L(\mathbf{p}_b, \gamma) = \sum_{c_j \in \mathcal{N}_b(c_i)}(p_j - \frac{1}{n_b})^2 + \sum_{c_j \in \mathcal{C} \setminus \mathcal{N}_b(c_i)}(0 - \frac{1}{n_b})^2 - \gamma(\sum_{c_j \in \mathcal{N}_b(c_i)} p_j - 1)$, where γ is called the Lagrange multiplier [36]. To proceed, we take the partial derivatives of the Lagrangian with respect to γ and p_j for all

$c_j \in \mathcal{N}_b(c_i)$, as

$$L_\gamma(\mathbf{p}_b, \gamma) = \sum_{c_j \in \mathcal{N}_b(c_i)} p_j - 1 = 0, \text{ and} \tag{3.21}$$

$$L_{p_j}(\mathbf{p}_b, \gamma) = 2\left(p_j - \frac{1}{n_b}\right) - \gamma = 0, \quad \forall c_j \in \mathcal{N}_b(c_i). \tag{3.22}$$

By plugging (3.22) into (3.21), we get $\frac{\gamma}{2} + \frac{1}{n_b} = \frac{1}{d_b(c_i)+1}$. Substituting this back to (3.22) obtains Theorem 4. □

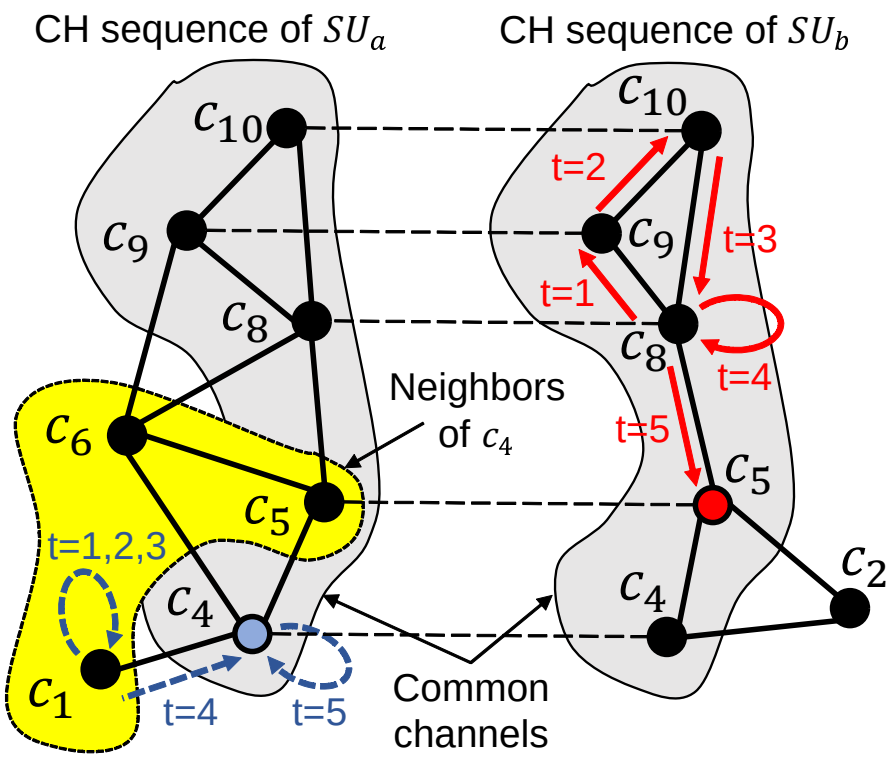
A *naive* random-based approach using Algorithm 3 can simply select the next-hop channel c_j uniformly at random among the neighbors of the current channel c_i in the spectrum slice graph. Theorem 1, on the other hand, suggests that the current channel c_i must also be able to be re-selected as the next-hop channel, for optimal MTTR. Specifically, SU_b must select both the neighbors of c_i and c_i itself with a uniform probability. For example, SU_b with a spectrum slice graph $\mathcal{G}_b$ depicted in Fig. 3.6a that is currently located at c_8 has to hop to either c_5, c_9, c_{10} or c_8 itself, each with probability $\frac{1}{d_b(c_8)+1} = \frac{1}{4}$. We will show in Section 3.5 that the optimal solution in Theorem 4 will result in a smaller rendezvous delay than the aforementioned naive approach.

3.3.4 PBNS-Assisted R-RCH for Quicker Rendezvous

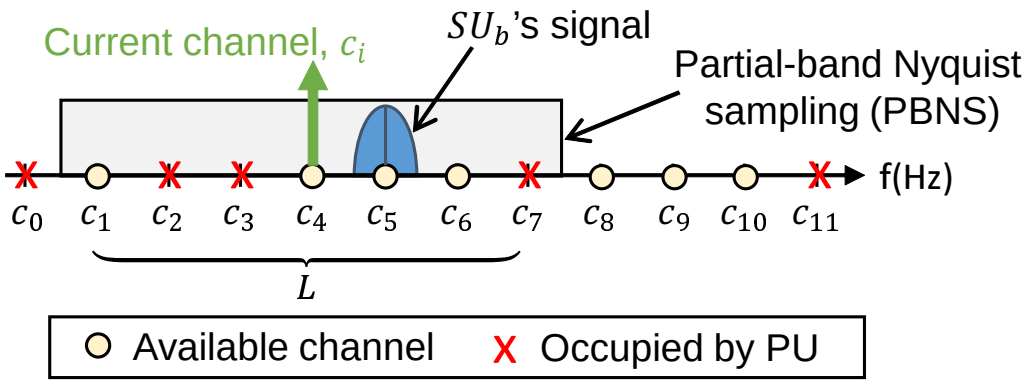
According to Definition 11, rendezvous may only occur if the walks (CH sequences) performed by nodes SU_a and SU_b visit the same vertex (channel). In this subsection, we show that the nodes can still achieve rendezvous even if they hop to two different, but neighboring vertices.

Consider an R-RCH scheme in which the CH sequences in Fig. 3.6b are re-drawn into Fig. 3.7a. We consider time slot $t = 5$ and focus on the child node SU_a, which visits the vertex c_4 in its spectrum slice graph. Instead of checking the mother node's (SU_b's) arrival at vertex c_4 only, SU_a can also "query" the neighbors, $\mathcal{N}_a(c_4) = \{c_1, c_5, c_6\}$ (see the yellow area with dotted boundary), to check whether they are visited by SU_b. Since at the same time SU_b hops to c_5, then the corresponding vertex reports to the child node and the node concludes there is a rendezvous.

The question is, how to query the neighboring vertices? To answer this question, we assume that every time the mommy node visits to a new channel, it transmits a pilot signal through the channel to indicate visitation. To listen for mommy, the child node SU_a can apply PBNS sensing [142]. The main benefit of using PBNS lies on its capability to probe pilot signals on SU_a's current channel, as well as the other ones within $\frac{L}{2}$ index away from the current channel (see Fig. 3.7b). Thus, applying PBNS becomes equivalent to checking the current channel as well as querying the neighboring channels.



(a) CH sequences of SU_a and SU_b up to time $t = 5$.



(b) PBNS sensing's capability to query neighboring channels.

Figure 3.7: An illustration of PBNS-assisted R-RCH scheme at time $t = 5$.

After being informed by the mommy node's presence, SU_a can quickly switch to mommy's channel, send acknowledgment signals, and proceed to control channel establishment. We refer the aforementioned scheme as a *PBNS-assisted R-RCH*. In the PBNS-assisted R-RCH scheme,

rendezvous can be achieved at time t^* if

$$X_a^{ch}(t^*) = c_{i_a}, X_b^{ch}(t^*) = c_{i_b}, \text{ and } c_{i_a} \in \mathcal{N}_a(c_{i_b}). \tag{3.23}$$

In Section 3.2.4, we assumed all IoT nodes have identical CH capability limit. Hence, the last constraint in (3.23) is equivalent to requiring $c_{i_b} = \mathcal{N}_b(c_{i_a})$, i.e., the current-hop channels of SU_a and SU_b are neighbors in spectrum graph of node SU_b.

Compared to the example in Fig. 3.6b, the PBNS-assisted scheme in Fig. 3.7a achieves rendezvous earlier at time $t = 5$. We will show in Section 3.5 that, in general, PBNS-assisted R-RCH can improve the TTR performance of R-RCH without PBNS assistance.

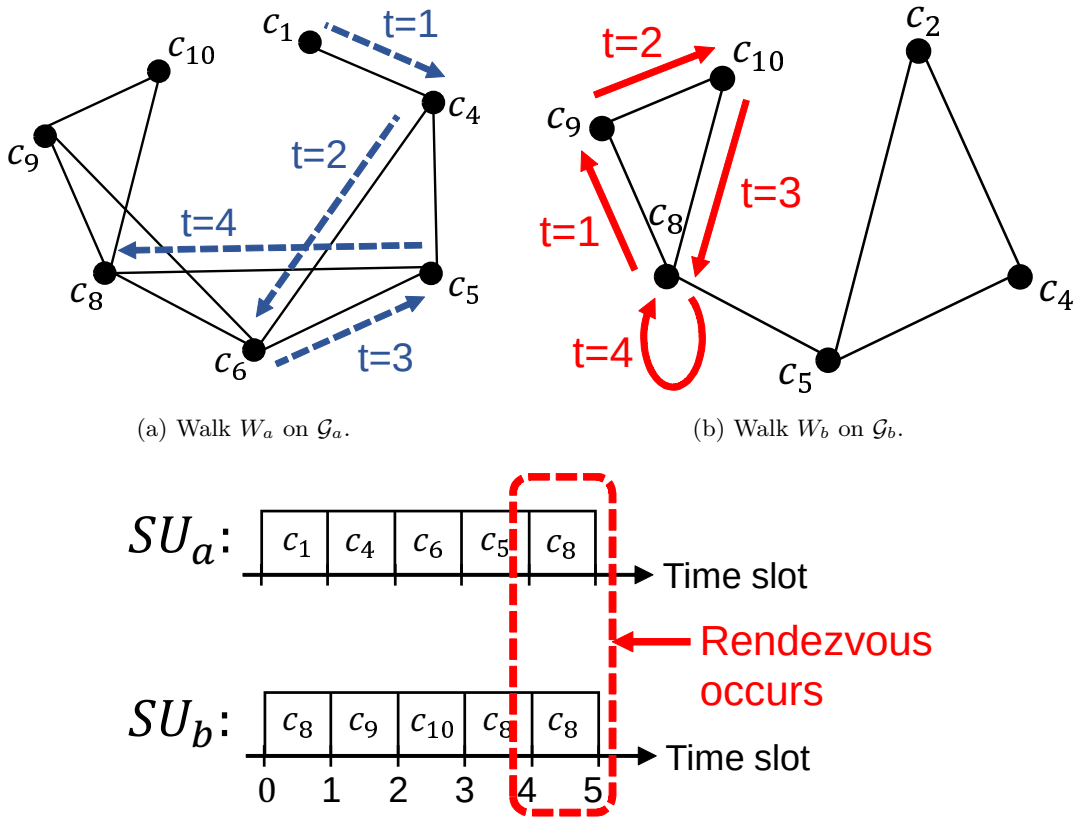
3.4 Symmetric Random Channel Hopping (S-RCH)

We have discussed how to achieve a quick rendezvous in DSA networks with pre-determined "child" (client) and "mother" (server) roles. Unlike the aforementioned hierarchical client-server setup, however, future IoT networks may have a flat peer-to-peer architecture, instead. For example, mesh-based online messaging apps, such as FireChat [2] and Jott [37], have enabled millions of proximate mobile devices to build a short-range wireless ad-hoc network for exchanging multimedia contents without requiring connection to a centralized server. Another example is vehicle-to-vehicle communications without infrastructure support [69], which enables the exchange of safety messages among nearby vehicles to promote safe driving and reduce traffic accidents. In these peer-to-peer-like IoT applications, the nodes do not have distinct pre-determined roles prior to rendezvous, so that a scheme where nodes have *symmetric* roles is desirable. In this section, we outline how to achieve such a rendezvous objective by proposing a *symmetric random CH* (S-RCH) scheme.

3.4.1 Preliminaries on S-RCH Scheme

Our main objective is to design a symmetric CH sequence where two IoT nodes, SU_a and SU_b, are assigned with identical CH processes. According to the WFM-based approach in Section 3.3, there are two possible alternatives: child-like or mommy-like CH process. When child-like CH is applied to both SU_a and SU_b, the resulting CH sequence will be almost static, thus inducing a very long rendezvous delay or even no rendezvous at all. To avoid long rendezvous, which contradicts our main goal of minimizing MTTR, we apply mommy-like CH process to both of the IoT nodes. The resulting scheme is what we refer as S-RCH.

In S-RCH, the CH sequences performed by SU_a and SU_b are modeled as random walks over their respective spectrum graphs. According to Definition 10, the walks are denoted as W_a and W_b, respectively. Denote $n_a := |\mathcal{C}_a|$ and let $\mathbf{P}_a$ be the $n_a \times n_a$ transition probability matrix of



(a) Walk W_a on $\mathcal{G}_a$.

(b) Walk W_b on $\mathcal{G}_b$.

(c) CH sequence versus time.

Figure 3.8: Illustration of symmetric random channel hopping (S-RCH) scheme.

walk W_a, where the (i, j)-th element of $\mathbf{P}_a$ is equal to the transition probability

$$P_a(c_i, c_j) := \Pr\{X_a(t+1) = c_j | X_a(t) = c_i\}. \tag{3.24}$$

The transition probability matrix $\mathbf{P}_b$ of walk W_b, on the other hand, follows the definition in (3.9).

We illustrate in Fig. 3.8 an S-RCH corresponding to the spectrum slice graphs in Fig. 3.1. In the examined S-RCH, walk W_a is illustrated as the dotted blue arrows in Fig. 3.8a, while the solid red lines in Fig. 3.8b belong to walk W_b. By plotting the CH sequence versus time in Fig. 3.8c, we observe that the rendezvous in S-RCH occurs on channel c_8 at time $t = 4$, which is earlier than $t = 10$ for the R-RCH in Fig. 3.6b.

Notice that S-RCH can be viewed as a special case of R-RCH where the child node, SU_a, hops every $M_s = 1$ time slot. Due to the singular M_s, the first term of the MTTR bound in

(3.12) will disappear so that a lower (better) MTTR can be expected. In the following, we will show how to minimize the MTTR of the S-RCH scheme by designing the transition probability matrices, $\mathbf{P}_a$ and $\mathbf{P}_b$, and then show that S-RCH achieves a lower MTTR than the R-RCH (without PBNS assist) when M_s is large.

3.4.2 S-RCH with Quick Rendezvous

We have discussed that S-RCH can be viewed as two random walks, denoted as W_a and W_b, over the spectrum slice graphs. In this section, we design the transition probability matrices of the random walks to achieve our main goal of minimizing MTTR. We have the following main result.

Theorem 5. *The transition probabilities of walks W_a and W_b that achieve the smallest MTTR satisfy*

$$P_a(c_i, c_j) = \begin{cases} \frac{1}{d_a(c_i)+1}, & \text{if } e(c_i, c_j) \in \mathcal{E}_a \text{ or } c_i = c_j, \\ 0, & \text{otherwise}, \end{cases} \tag{3.25}$$

$$P_b(c'_i, c'_j) = \begin{cases} \frac{1}{d_b(c'_i)+1}, & \text{if } e(c'_i, c'_j) \in \mathcal{E}_a \text{ or } c'_i = c'_j, \\ 0, & \text{otherwise}, \end{cases} \tag{3.26}$$

$\forall c_i, c_j \in \mathcal{C}_a$ *and* $\forall c'_i, c'_j \in \mathcal{C}_b$.

Before proving Theorem 5, we present the following general definition regarding random walks.

Definition 14. *A random walk W over graph $\mathcal{G} = (\mathcal{C}, \mathcal{E})$ with transition probability matrix $\mathbf{P}$ and stationary distribution vector $\boldsymbol{\pi}$ is*

1. *reversible if $\pi_{c_i} P(c_i, c_j) = \pi_{c_j} P(c_j, c_i), \forall c_i, c_j \in \mathcal{C}$,*
2. *irreducible if $\exists t < \infty$ such that $P^t(c_i, c_j) > 0$ for any $c_i, c_j \in \mathcal{C}$, and*
3. *aperiodic if $\exists N < \infty$ such that $\forall c_i \in \mathcal{C}, P^n(c_i, c_i) > 0$ for all $n \geq N$.*

Definition 14 can be applied to our walks-of-interest, W_a and W_b, by adding subscripts $()_a$ and $()_b$, respectively. Typically, the second and third conditions in Definition 14 are satisfied by W_a and W_b since their spectrum slice graphs, $\mathcal{G}_a$ and $\mathcal{G}_b$, are connected and non-bipartite.[6] On the other hand, many variants of random walk over graph [70] can be designed to satisfy the

[6] A graph is *bipartite* if the vertices can be divided into two disjoint sets and no two vertices in the same set are adjacent.

first condition in Definition 14. In fact, we can show that W_a and W_b employing the transition probabilities in (3.25)-(3.26) have the stationary distributions of

$$\pi^a_{c_i} := [\boldsymbol{\pi}_a]_i = \frac{d_a(c_i)+1}{2|\mathcal{E}_a|+|\mathcal{C}_a|} \text{ and} \tag{3.27}$$

$$\pi^b_{c'_i} := [\boldsymbol{\pi}_b]_i = \frac{d_b(c'_i)+1}{2|\mathcal{E}_b|+|\mathcal{C}_b|}, \tag{3.28}$$

which satisfy the reversibility (first) condition. Thus, it is reasonable to assume that W_a and W_b satisfy all the three conditions in Definition 14.

By considering W_a and W_b as two independent walks performed in parallel by nodes SU_a and SU_b, the current locations of the walks can be described by a two-variate state $(c_i, c'_i) := (X^{ch}_a(t) = c_i, X^{ch}_b(t) = c'_i)$. Furthermore, by denoting $(c_j, c'_j) := (X^{ch}_a(t+1) = c_j, X^{ch}_b(t+1) = c'_j)$ as the state at the next time slot, the transition probabilities of the individual walks can be combined into a two-variate transition probability

$$P_{ab}((c_i, c'_i), (c_j, c'_j)) = P_a(c_i, c_j) \times P_b(c'_i, c'_j), \tag{3.29}$$

where subscript $()_{ab}$ indicates that P_{ab} corresponds to the "parallel" walks consisting both W_a and W_b. Let $n_b := |\mathcal{C}_b|$ be the number of channels available to node SU_b. Then, we can state the $(n_a n_b) \times (n_a n_b)$ transition probability matrix of the parallel walks as

$$\begin{aligned} \mathbf{P}_{ab} &= \begin{bmatrix} P_a(c_{i_1}, c_{i_1})\mathbf{P}_b & \dots & P_a(c_{i_1}, c_{i_{n_a}})\mathbf{P}_b \\ \vdots & \ddots & \vdots \\ P_a(c_{i_{n_a}}, c_{i_1})\mathbf{P}_b & \dots & P_a(c_{i_{n_a}}, c_{i_{n_a}})\mathbf{P}_b \end{bmatrix} \\ &= \mathbf{P}_a \otimes \mathbf{P}_b, \end{aligned} \tag{3.30}$$

where $\otimes$ denotes the Kronecker matrix operator.

To proceed, we study several properties regarding the parallel walks, which for simplicity is denoted as W_{ab}. Let $\boldsymbol{\pi}_a$ and $\boldsymbol{\pi}_b$ be the stationary distribution vectors of $\mathbf{P}_a$ and $\mathbf{P}_b$, respectively. Denote $\boldsymbol{\pi}_{ab}$ as the $(n_a n_b) \times 1$ stationary distribution vector corresponding to the transition probability matrix $\mathbf{P}_{ab}$, which satisfies $\boldsymbol{\pi}_{ab}\mathbf{P}_{ab} = \mathbf{P}_{ab}$. Moreover, let the mixing time of the parallel walks W_{ab} be denoted as

$$\begin{aligned} &\tau^{ab}_{mix}(\epsilon) \\ &:= \inf\left\{ t : \max_{c_i \in \mathcal{C}_a, c'_i \in \mathcal{C}_b} \left| (P^t(\{c_i, c'_i\}, \{\cdot\}) - \pi_{\{\cdot\}}) \right| \le \epsilon \right\}. \end{aligned} \tag{3.31}$$

Then, we have the following lemma.

Lemma 5. *The MTTR of the S-RCH scheme is bounded by*

$$TTR_{max} \geq \frac{1}{2}\tau_{mix}^{ab}(1/4) - \frac{1}{2}. \tag{3.32}$$

Proof. We start by adopting the notion of *maximum hitting time*

$$t_{Hit} := \max_{c_i,c_j \in \mathcal{C}_a, c_i', c_j' \in \mathcal{C}_b} E_{\{c_i,c_i'\}}[\tau_{\{c_j,c_j'\}}] \tag{3.33}$$

from [100], where the *first hitting time*

$$\tau_{(c_j,c_j')} := \min\{t \geq 0 : X_a(t) = c_j, X_b(t) = c_j'\} \tag{3.34}$$

denotes the first time state $\{X_a(t), X_b(t)\} = \{c_j, c_j'\}$ is visited by the walk. Moreover, we know that the maximum hitting time is bounded by [100, Theorem 10.14]

$$t_{Hit} \geq \frac{1}{2}\tau_{mix}^{ab}(1/4) - \frac{1}{2}. \tag{3.35}$$

Let the *largest first hitting time* be denoted as $\tau_{(c_j,c_j')}^{max} := \max_{c_j \in \mathcal{C}_a, c_j' \in \mathcal{C}_b} \tau_{\{c_j,c_j'\}}$. We observe that

$$TTR_{max} \geq \tau_{(c_j,c_j')}^{max} \geq t_{Hit}, \tag{3.36}$$

where the first inequality on the left side holds because the MTTR (TTR_{max}) requires an additional condition $X_a(t) = X_b(t) = c^*$ to be satisfied (see (3.5)-(3.6)) and is more restricted that the largest first hitting time. The second inequality in (3.36), on the other hand, holds by the relation between the maximum and expected values. Finally, combining (3.36) with (3.35) completes the proof of Lemma 5. □

Lemma 5 reveals that MTTR is lower-bounded by a factor proportional to the mixing time $\tau_{mix}(\epsilon)$. So, minimizing the best-case MTTR is equivalent to minimizing $\tau_{mix}(\epsilon)$, and we will optimize $\tau_{mix}(\epsilon)$ to design the transition probability matrix $\mathbf{P}_{ab}$ that minimizes the MTTR. Before designing $\mathbf{P}_{ab}$, we outline several useful properties as follows.

Lemma 6. *The parallel walks with transition probability matrix* $\mathbf{P}_{ab} = \mathbf{P}_a \otimes \mathbf{P}_b$ *have a stationary distribution vector of* $\boldsymbol{\pi}_{ab} = \boldsymbol{\pi}_a \otimes \boldsymbol{\pi}_b$.

Lemma 7. *The parallel walks are reversible, irreducible, and aperiodic.*

With Lemmas 7 and 6 at hand, we show the uniqueness of the stationary distribution vector $\boldsymbol{\pi}_{ab}$ as follows.

Lemma 8. *The parallel walks have a unique stationary distribution vector* $\boldsymbol{\pi}_{ab} \geq \mathbf{0}$.

Proof. Since the parallel walks are reversible, irreducible, and aperiodic, it have a unique transition probability $\boldsymbol{\pi}_{ab}$ with $[\boldsymbol{\pi}_{ab}]_{(c_i,c_i')} > 0$ for all $c_i \in \mathcal{C}_a$ and $c_i' \in \mathcal{C}_b$ [146, Proposition 4]. □

Now we are ready to derive the mixing time upper and lower bounds that will be used for minimizing the MTTR in Lemma 5. Let $\lambda_a^* := \max\{\lambda_2^a, -\lambda_n^a\}$ be the SLEM associated with walk W_a, where λ_i^a is the i-th largest eigenvalues of $\mathbf{P}_a$. Moreover, let $\pi_a^* := \min_{c_i \in \mathcal{C}_a} \pi_{c_i}^a$ be the smallest element of the stationary distribution vector $\boldsymbol{\pi}_a$. Similar definition also applies for walk W_b by substituting a with symbol b.

Proposition 1. *The mixing time of the parallel walks* W_{ab} *is lower and upper bounded by*

$$\left(\frac{1}{1-\lambda_a^*\lambda_b^*} - 1\right) \log\left(\frac{1}{2\epsilon}\right) \leq \tau_{mix}^{ab}(\epsilon) \leq \frac{1}{1-\lambda_a^*\lambda_b^*} \log\left(\frac{1}{\epsilon \times \pi_a^* \pi_b^*}\right). \tag{3.37}$$

Proof. From Lemma 7, we know that the walk W_{ab} with transition probability matrix $\mathbf{P}_{ab}$ is reversible, irreducible, and periodic. Thus, according to (3.17), W_{ab} has the mixing time bounded by $\left(\frac{1}{1-\lambda^*(\mathbf{P}_{ab})} - 1\right) \times \log(\frac{1}{2\epsilon}) \leq \tau_{mix}^{ab}(\epsilon) \leq \frac{1}{1-\lambda^*(\mathbf{P}_{ab})} \times \log\left(\frac{1}{\epsilon \times \pi_{min}(\mathbf{P}_{ab})}\right)$, where $\lambda^*(\mathbf{P}_{ab})$ and $\pi_{min}(\mathbf{P}_{par})$ are respectively the SLEM and the smallest entry of $\boldsymbol{\pi}_{ab}$. First, we focus on the common term, $\frac{1}{1-\lambda^*(\mathbf{P}_{ab})}$. According to [82, Theorem 4.2.12], we know that $\lambda_i^a \times \lambda_j^b$ for any $i \leq n_a$ and $j \leq n_b$ is also an eigenvalue of $\mathbf{P}_{ab}$, such that the spectral gap can be re-stated as $1 - \lambda^*(\mathbf{P}_{ab}) = \inf_{2 \leq i \leq n_a, 2 \leq j \leq n_b} 1 - |\lambda_i^a||\lambda_j^b|$.

Next, we consider the $\frac{1}{\epsilon \times \pi_a \pi_b}$ term in the upper bound. According to Lemma 6, $\pi_{c_i}^a \times \pi_{c_i'}^b$ for any $c_i \in \mathcal{C}_a$ and $c_i' \in \mathcal{C}_b$ is also a stationary distribution corresponding to $\mathbf{P}_{ab}$, and $\boldsymbol{\pi}_{ab} = \boldsymbol{\pi}_a \otimes \boldsymbol{\pi}_b$ is unique by Lemma 8. Thus, the logarithm term in the upper bound can be obtained by taking the minimum over all possible $c_i \in \mathcal{C}_a$ and $c_i' \in \mathcal{C}_b$, completing the proof. □

With Proposition 1 and Lemma 5 at hand, we are now ready to prove Theorem 5 as follows.

Proof of Theorem 5. Lemma 5 reveals that minimizing MTTR is equivalent to minimizing the mixing time $\tau_{mix}^{ab}(1/4)$, which according to Proposition 1 is proportional to $\lambda_a^* \times \lambda_b^*$. Note that λ_a^* and λ_b^* are the SLEMs contributed by walks W_a and W_b, respectively, while the walks are operated independently to each other. As a result, minimizing $\lambda_a^* \times \lambda_b^*$ can be done by separately minimizing λ_a^* and λ_b^*. To this end, Thoerem 4 indicates that a walk that minimizes its SLEM λ^* must satisfy the transition probabilities in (3.8). As a result, to minimize λ_a^* and λ_b^* as well as the MTTR, we leverage the result in (3.8) and add the appropriate subscripts for W_a and W_b to obtain Theorem 5. □

3.4.3 Comparison Between R-RCH and S-RCH Schemes

The main difference between the symmetric (S-RCH) and role-based (R-RCH) schemes is that the node SU_a needs to select and hop to a new channel more often in the former. In this subsection, we will leverage our mathematical analysis to show that the increased complexity imposed by S-RCH is traded off with lower (better) MTTR.

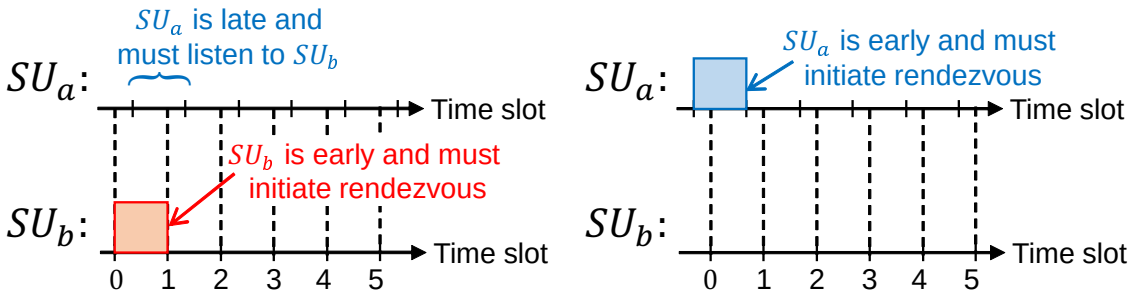
To highlight the performance advantage of the S-RCH scheme, we consider a *homogeneous* DSA network where IoT nodes SU_a and SU_b observe the same available channels, $\mathcal{C}_a = \mathcal{C}_b$. When the network is homogeneous, the analysis for parallel walks (S-RCH) can also be applied to analyze an R-RCH scheme with $M_s \to \infty$, that is, when node SU_a stays at a channel indefinitely. In this case, the transition probability matrix of walk W_a by SU_a is equal to $\mathbf{P}_a = \mathbf{I}$ while the SLEM is now given as $\lambda_a^* = 1$. From the parallel walk's perspective, the combined SLEM of the role-based (R-RCH) scheme becomes $\lambda_a^* \times \lambda_b^* = \lambda_b^*$. The SLEM of the symmetric (S-RCH) scheme, on the other hand, is equal to $\lambda_a^* \times \lambda_b^*$ with $\lambda_a^* \leq 1$. Since $\lambda_b^* \geq \lambda_a^* \times \lambda_b^*$ and the combined SLEM is proportional to the MTTR (see Proposition 1 and Lemma 5), we conclude that in homogeneous networks with $M_s \to \infty$, the R-RCH scheme has a larger (worse) MTTR than the S-RCH scheme. In Section 3.5.3, we will show that a similar result can be observed in heterogeneous networks with finite M_s.

3.4.4 PBNS-Assisted S-RCH Scheme

Similar to the role-based (R-RCH) scheme in Section 3.3.4, the rendezvous delay of S-RCH can also be shortened by allowing the SU nodes to rendezvous on two different but neighboring vertices (channels) in the spectrum slice graph. The approach in Section 3.3.4 that asymmetrically assigns the PBNS task to node SU_a only, however, cannot be applied to S-RCH in which every node must be assigned a symmetric (identical) role. To this end, we leverage the practical assumption that SU_a and SU_b may start the CH process at different times due to their unsynchronized local clocks, and apply the following *listen-before-transmit* (LBT) strategy.

1. *Listen phase:* Firstly, the node SU_a performs a PBNS with range L centered at its current channel $X_a^{ch}(t)$. If there is a pilot signal detected, e.g., the blue tone transmitted by SU_b at c_5 (see Fig. 3.7b), then SU_a decodes the signal, waits until the signal ends, and transmits an acknowledgment (ACK) to indicate a successful rendezvous. Otherwise, proceed to the phase below.

2. *Transmit phase:* Secondly, SU_a hops to a new channel $X_a^{ch}(t)$ selected according to Theorem 5, transmits a pilot signal, and wait for and ACK from SU_b. If an ACK is received, then stop because rendezvous had occurred. Otherwise, increment time t and go to the listen phase to attempt another rendezvous.



(a) Listen when clock is late.

(b) Transmit when clock is early.

Figure 3.9: Node SU_a's action depending on its local clock's position.

The listen phase is for the case that SU_a's local clock is later than SU_b (see Fig. 3.9a), to make sure that SU_a can hear the rendezvous signal transmitted by SU_b. On the other hand, the transmit phase is for accounting when SU_a has an early local clock and has to initiate rendezvous (see Fig. 3.9b). In this paper, we refer an S-RCH applying the aforementioned LBT phases as a *PBNS-assisted S-RCH* scheme.

3.5 Performance Evaluation

After outlining and discussing how to achieve quick rendezvous via role-based (R-RCH) and symmetric (S-RCH) channel hopping schemes, we evaluate their performance via numerical and experimental evaluations. In our evaluations, we consider the following network parameters to build the spectrum slice graphs $\mathcal{G}_a$ and $\mathcal{G}_b$ for rendezvous, according to the method outlined in Sections 3.2.3 and 3.2.4.

3.5.1 Parameter Setup

We consider LTE-U [65], which is a recent standard that enables opportunistic wireless access via DSA over the following channels: U-NII-1 (over the 5.25-5.33 GHz frequencies), U-NII-2C (on the 5.49-5.73 GHz spectrum), and U-NII-3 (covering the 5.735-5.835 GHz bands). Thus, the considered DSA network covers a 500MHz spectrum in total. We focus on future LTE-U networks with SUs having $B_0 = 100$ kHz channel raster[1] [65] for efficient spectrum usage under dense IoT-based deployment, which results in $N = 5000$ non-overlapping channels. To ensure consistency with the LTE frames used in LTE-U standard, each IoT node applies a uniform time slot period of $\delta = 10$ ms.[2] Moreover, each IoT node is assumed to carry an on-board SDR with similar specifications to USRP E310, which is a commercial of-the-shelf SDR with embedded ARM processor, resembling those found in modern smartphones and tablets. As a

result, we assume that the SDR has a digital bandwidth of $BW_{SDR} = 56MHz$ and the resulting CH range becomes limited to $R_{ch} = 139$ (see eqs. (3.1)-(3.3) for the relation between R_{ch} and BW_{SDR}).

In the considered LTE-U network, there are two types of PUs: 1) 5GHz IEEE 802.11 WiFi devices, which reserves the U-NII-1 and U-NII-3 bands, as well as 2) weather and military radars in the U-NII-2C bands. There are a total of 25 U-NII channels that can be occupied by PU devices, each with a bandwidth of $BW_{pu} = 20$ MHz. Note that since the SUs' bandwidth of $B_0 = 100$ kHz is much smaller than BW_{pu}, each unoccupied U-NII (PU) channel consists of $\frac{BW_{pu}}{B_0} = 200$ SU channels for the IoT nodes. We assume a network with moderate PU activities, where the number of unoccupied U-NII channels is determined as follows. Let an *overlap ratio* $p_{ab} := \frac{|\mathcal{C}_a \cap \mathcal{C}_b|}{N}$ be the fraction of channels available to IoT nodes SU_a and SU_b. Unless specified otherwise, p_{ab} is set to 0.2. Moreover, let $p_a := \frac{|\mathcal{C}_a \setminus (\mathcal{C}_a \cap \mathcal{C}_b)|}{N}$ and $p_b := \frac{|\mathcal{C}_b \setminus (\mathcal{C}_a \cap \mathcal{C}_b)|}{N}$ be the *fraction of non-overlap* U-NII channels available to nodes SU_a and SU_b, respectively, that have a default value of $p_a = p_b = 0.2$. According to these fractions, the sets of unoccupied U-NII channels are chosen randomly from all the 25 U-NII channels and remain fixed throughout each network realization. Then, the number of unoccupied U-NII channels around SU_a is given as $25 \times (p_a + p_{ab})$, while that of node SU_b is $25 \times (p_b + p_{ab})$.

In our evaluation, we compare the proposed R-RCH scheme employing the optimal transition probabilities in Theorem 4 (denoted as `R-RCH (opt)`) to the same scheme applying a simple random walk [100] (`R-RCH (SRW)`), which represent a naive random-based scheme. Moreover, we evaluate the proposed S-RCH scheme (`S-RCH (opt)`) and compare it to the Modular Modified Clock scheme [143] (`MMC`), which is a state-of-the-art randomized scheme. The results for PBNS-assisted schemes in Sections 3.3.4 and 3.4.4 are also included for comparison. All the data points in our results are averaged over 10^4 measurement runs.

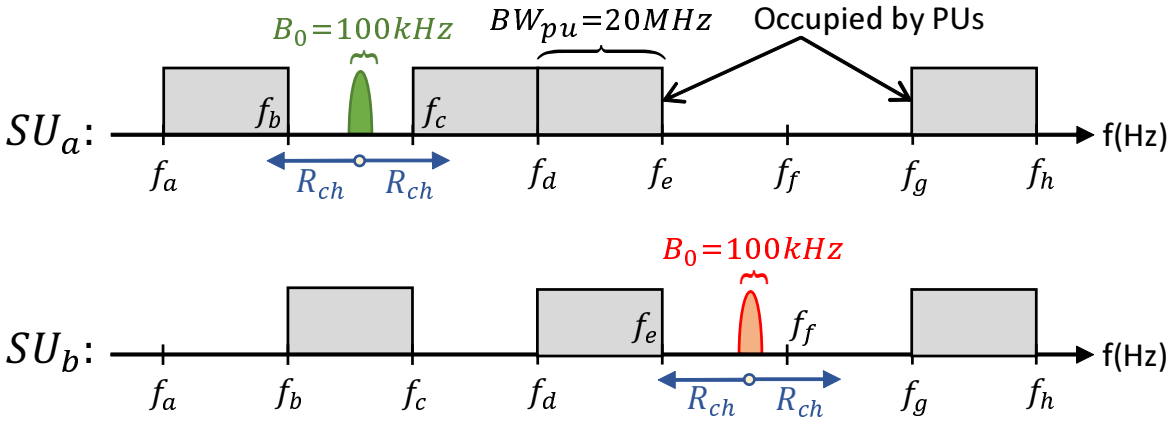


Figure 3.10: Local optima problem due large PU bandwidth and small CH range.

3.5.2 Avoiding Local Optima

The wide U-NII channel bandwidth of 20MHz—each providing $N = 200$ SU channels—combined with the narrow CH range limit (e.g., $R_{ch} = 139$ when USRP E310 SDR is used) may prevent SU_a and SU_b from achieving rendezvous. To this end, multiple PUs may occupy consecutive spectrum bands (i.e., at frequencies $[f_c, f_d]$ and $[f_d, f_e]$ near SU_a), as illustrated in Fig. 3.10. Consider the node SU_a. The limited CH range R_{ch} prevents SU_a from hopping beyond the 400 occupied SU channels on frequencies $[f_c, f_e]$ and traps SU_a's CH sequence within a *local optima*[7] (in this case, frequencies f_b to f_c). The CH sequence of SU_a trapped in a local optima is equivalent to having a random walk over a disconnected spectrum slice graph $\mathcal{G}_a$, where the walk is confined to a locally connected component. As a result, SU_a may not be able to hop to the frequency of SU_b (i.e., between $[f_e, f_d]$ in Fig. 3.10) and rendezvous will never happen. In order to prevent the IoT-based SU nodes from being trapped in a local optima, we let the nodes to select an arbitrary channel every $T_{reset} = 1000$ time slots. When the selected channel is occupied by PU, the corresponding node will repeat the selection until a vacant channel is picked.

3.5.3 Time-to-Rendezvous

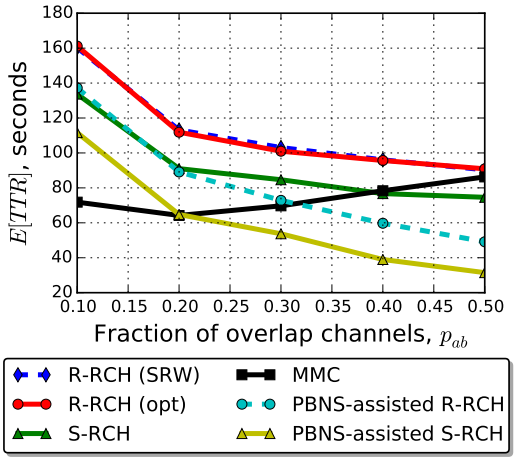
We select the expected TTR—the average delay until nodes SU_a and SU_b can achieve rendezvous—as the main metric-of-interest. Then, the following scenarios are evaluated.

Effect of the fraction of overlap channels (p_{ab})

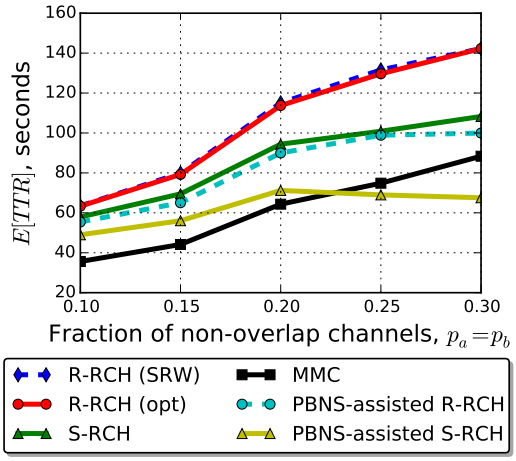
We consider the effect of increasing p_{ab} to the expected TTR by presenting the corresponding plots in Fig. 3.11a. As p_{ab} becomes higher, rendezvous is more likely to occur early since there will be more commonly-available channels to both SU_a and SU_b, and is verified by the decreasing expected TTR of the role-based `R-RCH (SRW)` scheme (see the upper-most dotted line in blue). The proposed `R-RCH (opt)` scheme, on the other hand, achieves an improved expected TTR since it attempts to minimize the MTTR bound in Lemma 3, by applying the transition probabilities in Theorem 4 to the "mother" node, SU_b. When the symmetric (`S-RCH`) scheme is employed, both SU_a and SU_b actively attempt to rendezvous by hopping to a newly-selected channel at each time slot. Such a two-way effort provides a lower (better) expected TTR than the role-based (R-RCH) schemes, which essentially involve a one-sided effort by node SU_b only.

Fig. 3.11a also shows that the PBNS-assisted R-RCH and S-RCH schemes, which allows the IoT SU nodes to detect rendezvous signals outside their current channels as long as the

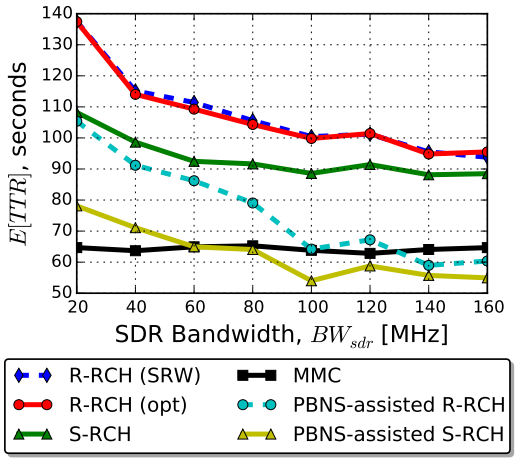
[7]The local optima term, borrowed from convex optimization [36], refers to a solution to an optimization problem over a limited neighborhood (range) that is not necessarily a globally-optimum solution.



(a) Versus fraction of overlap channels.



(b) Versus fraction of non-overlap channels.



(c) Versus SDR's bandwidth.

Figure 3.11: Expected TTR performance with respect to various network parameters.

condition in (3.23) is satisfied, can reduce the expected TTR (see the cyan and yellow plots on the bottom). These results indicate that although the PBNS sensing causes the CH range limit R_{ch} in (3.2)-(3.3), it can actually be leveraged to improve the rendezvous delay performance.

We additionally compare the performance of the proposed schemes to that of the existing `MMC` scheme [143]. Since `MMC` has a CH sequence that is designed without accounting for CH capability limits, it requires the SDR to be able to hop to any arbitrary channel, irrespective of the frequency difference to the current channel. As a result, the SDR needs to do a two-stage tuning at each hop (recall Section 3.2.3 for details), which takes at least 30ms for practical IoT-based embedded application [26]. To provide additional time for transmitting and/or decoding rendezvous signals, we set the `MMC` scheme's time slot to $4\delta = 40$ms. The corresponding result

for MMC (see the solid black plot in Fig. 3.11a) indicates that MMC outperforms the proposed schemes without PBNS-assist when p_{ab} is low. As p_{ab} is increased to 0.5, the performance of MMC worsens and converges to the role-based (R-RCH) scheme. The PBNS-assisted R-RCH and S-RCH schemes, on the other hand, generally outperforms the MMC scheme with up to 70.2% expected TTR improvement.

Effect of the fraction of non-overlap channels (p_a and p_b)

We consider the effect of the fraction of non-overlap channels by setting $p_a = p_b$ and increasing their values from 0.1 to 0.3 in Fig. 3.11b. As p_a and p_b increases, the random walks performed by SU_a (in S-RCH and MMC schemes) and SU_b (in all schemes) effectively spend more time in the non-overlap channels than in the common channels, $\mathcal{C}_a \cap \mathcal{C}_b$. Since rendezvous in these schemes only happen in $\mathcal{C}_a \cap \mathcal{C}_b$, the likelihood of rendezvous at each time slot is diminished and the expected TTR becomes larger. Then, the expected TTR becomes an increasing function of p_a and p_b, as indicated by the plots in Fig. 3.11b.

Effect of SDR's bandwidth (BW_{SDR})

Next, we increase the SDR's bandwidth, BW_{SDR}, from 20MHz up to 160MHz[3] and plot the resulting expected TTR in Fig. 3.11c. According to (3.1)-(3.3), BW_{SDR} is proportional to the CH capability limit R_{ch}, while higher R_{ch} is helpful when the IoT SU nodes start in two channels with vastly different indexes (e.g., $X_a(0) = c_0$ and $X_b(0) = c_{N-1}$). Specifically, by assuming an imaginary *best-case* policy under the role-based CH scenario, node SU_a can stay put while SU_b constantly hops from c_i to $c_{i-R_{ch}}$ at each step, thus resulting in a TTR of $\lceil \frac{N}{R_{ch}} \rceil$. Additionally, SU_b can follow the same deterministic policy, which results in a combined TTR of $\lceil \frac{N}{2R_{ch}} \rceil$ in the *best-case* symmetric CH scenario. We found that these inversely-proportional scaling of TTR versus R_{ch} (and BW_{SDR}) can also be observed in the *average-case*, as demonstrated by the expected TTR plots in Fig. 3.11c. As a result, the expected TTR decreases and the performance of the proposed R-RCH and S-RCH schemes approach that of the MMC as BW_{SDR} becomes larger. The expected delay of MMC scheme that does not account for the limited CH capability, on the other hand, remains constant versus BW_{SDR}.

When PBNS-assisted schemes are employed and BW_{SDR} is increased, the range of sensible channels, L, grows larger. As a result, the maximum number of neighbors (i.e., the upper bounds of $\mathcal{N}_a$ and $\mathcal{N}_b$) increase and the rendezvous condition in (3.23) becomes looser. In other words, rendezvous occurs more easily and not only does the PBNS-assisted schemes offer improved expected TTR performance, but also their absolute performance gaps—with respect to R-RCH and S-RCH without PBNS assistance—becomes larger, as indicated by the cyan and yellow plots in Fig. 3.11c.

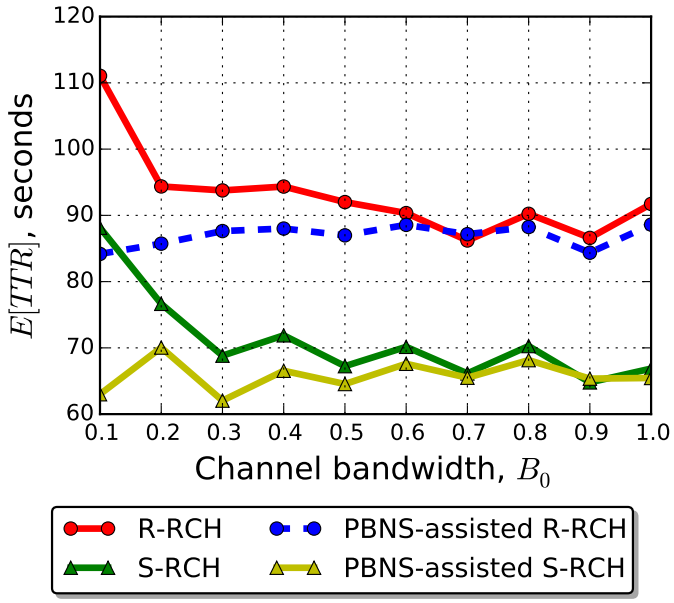


Figure 3.12: Expected TTR versus SU's bandwidth, B_0.

Remark 5. *Fig. 3.11b shows that the expected TTR is an increasing function of the portion of non-overlap channels, p_a and p_b, while Figs. 3.11a and 3.11c indicate that the TTR is inversely-proportional to the fraction of overlaps, p_{ab}, and the SDR's bandwidth, BW_{SDR}. From a hardware designer's perspective, it is desirable to increase BW_{SDR}, since the remaining variables (p_a, p_b, and p_{ab}) are intrinsic to the DSA network's operating environment. Increasing BW_{SDR}, however, trades improved TTR performance with greater device complexity and manufacturing cost.*

Effect of SU channel's bandwidth (B_0)

Some IoT applications such as environment monitoring involves a large number of sensor networks with small throughput requirement, thus ideally served with a small channel bandwidth B_0. On the other hand, large B_0 may be needed to support the exchange of large contents, such as 4K video recording, between users of mesh-based messaging apps. To understand how different values of B_0 affect rendezvous delay, we fix the overlap and non-overlap channel ratios (p_a, p_b, and p_{ab}), increase the SU's bandwidth from 0.5MHz to 5.0MHz, and then evaluate the corresponding expected TTR in Fig. 3.12. To this end, the total number of available SU channels, $|\mathcal{C}_a \cup \mathcal{C}_b| = \left\lfloor \frac{Total\_spectrum\_BW}{B_0} \right\rfloor (p_a + p_b + p_{ab})$, will decrease with respect to B_0. Since the CH schemes need to operate over a larger search space before rendezvous, the expected TTR will increase with respect to $|\mathcal{C}_a \cup \mathcal{C}_b|$ (equivalently, as B_0 becomes smaller), which is verified in

Fig. 3.12 for R-RCH and S-RCH schemes without PBNS assistance.

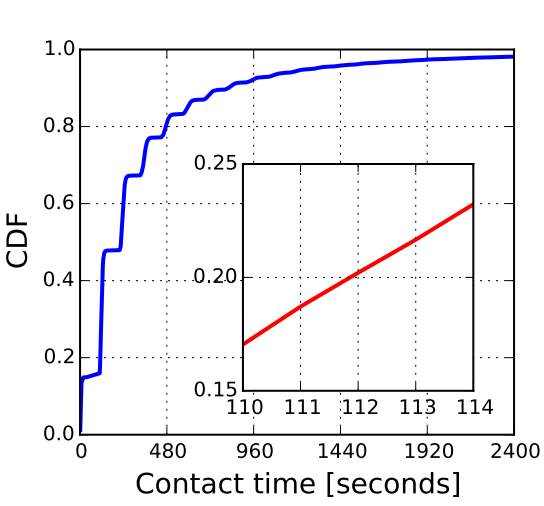
On the other hand, the expected TTR of the PBNS-assisted schemes are relatively constant versus B_0. The main reason is that the PBNS's capability of sensing pilot signals on other channels, $\lfloor \frac{L-1}{2} \rfloor$ (see the first condition in (3.23)), also decreases with respect to B_0. The resulting decline of the expected TTR then counter-balances the improvement provided by the reduced search space, thus resulting in a zero net improvement. Moreover, according to (3.2), the number of sensible channels L is inversely proportional to B_0. As a result, the expected TTR of PBNS-assisted R-RCH and S-RCH schemes converge to those without PBNS assistance since the rendezvous criterion in (3.23) becomes equivalent to that in Definition 11, as B_0 becomes very large.

3.5.4 Time-to-Rendezvous Versus IoT Nodes' Mobility

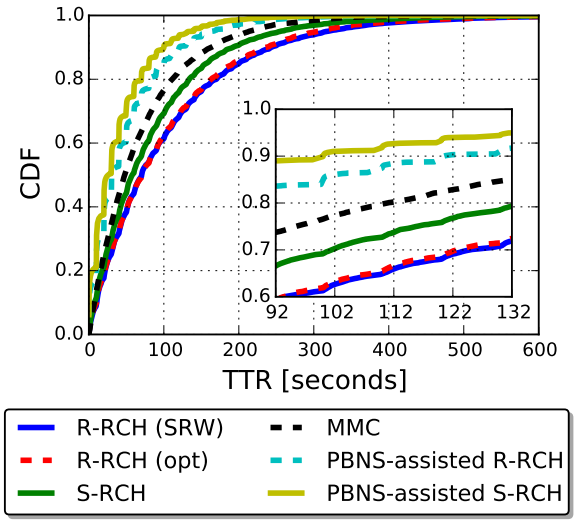
Modern IoT-based DSA networks, including mesh-based online messaging applications like FireChat and Jott, consist of mobile nodes with momentary inter-node contacts as their SUs. One major concern is whether the time delay for achieving pairwise rendezvous is sufficiently small and can exploit the opportunities provided by the momentary contacts. To examine the capability of the proposed CH schemes to facilitate rendezvous, we employ the `cambridge/haggle` data set [130] that records the pairwise Bluetooth sightings (contacts) by groups of nodes carrying small devices (iMotes) in indoor environments. Specifically, we consider the `Exp6` trace that collects the time and duration of contacts between iMotes distributed to 78 students attending the Infocom'06 conference between April 23 to 26, 2017 in the Princesa Sofia Gran Hotel, Barcelona. The `Exp6` trace is employed to portray future conference settings, in which there are many international participants carrying 5G handsets with SDR capability but not equipped with data roaming access, due to expensive roaming fee. In this case, the participants can use the SDR to access unused spectrum via DSA for communications.

We assume that the Bluetooth sightings provided by the `Exp6` data trace captures all the possible short-range physical contacts between mobile SUs. The *contact times*, which are denoted as $\{CT\}$ and quantify the duration from when a pair of nodes come into contact until they move out of each others' contacts, are collected. For practicality, we discard the results when the contact time is zero. The corresponding cumulative distribution function (CDF), $Pr\{CT < x\}$, is plotted in Fig. 3.13a. Note that there are irregularities (step-like profiles) about every 120s since the scanning intervals are chosen randomly from the (120 ± 12)s range [130].

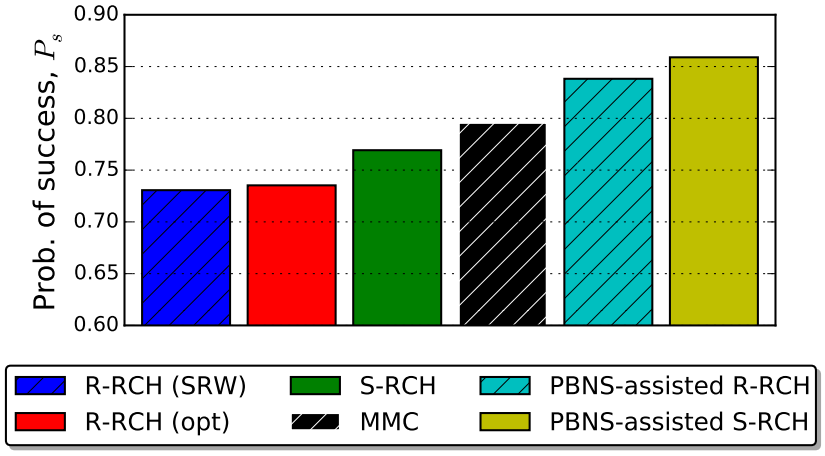
To proceed, we present Fig. 3.13b that plots the rendezvous delay (TTR) corresponding to the schemes evaluated in Section 3.5.3, and then relate the TTR plots to the CDF of contact times in Fig. 3.13a. To be successful, a rendezvous must occur within a contact period, i.e., when $TTR < CT$. Consider the CDF of the contact times in Fig. 3.13a when most (80%) of the



(a) Mobile SUs' contact times.



(b) Distribution of TTR.



(c) Probability of successful rendezvous.

Figure 3.13: Performance evaluation of the proposed CH schemes in mobile indoor environment [130].

contact times have not ended, that is, $CT > 112$ (see the red plot in the inset of Fig. 3.13a). Then, Fig. 3.13b indicates that the proposed `R-RCH (opt)`, `S-RCH`, `PBNS-assisted R-RCH`, and `PBNS-assisted S-RCH` schemes achieve successful rendezvous with probabilities 0.665, 0.706, 0.885, and 0.927, respectively.

Although possible, comparing Figs. 3.13a and 3.13b for each possible contact time and TTR—as in the example above—is cumbersome. To provide a more concise way for evaluating a rendezvous scheme's ability to overcome the node mobility, we denote a *successful rendezvous*

probability metric as follows.

$$P_s = Pr(TTR < CT) = \sum_{x=0}^{\infty} Pr(TTR < x) \times Pr(CT = x). \tag{3.38}$$

Metric P_s quantifies the likelihood that a successful rendezvous can be achieved, over all the possible contact times, CT. From the resulting P_s plots in Fig. 3.13c, we observe that the optimal transition probabilities applied by the proposed `R-RCH (opt)` scheme provide 0.64% higher success rate than the naive `R-RCH (SRW)` scheme and a 7.59% lower P_s than the existing `MMC` schemes. The performance of the proposed `S-RCH` scheme, on the other, is in-between the `R-RCH (opt)` and `MMC` schemes. Moreover, the proposed `PBNS-assisted R-RCH and S-RCH` schemes improve the successful rendezvous probability of the `MMC` by 5.35% and 7.95%, respectively. As a result, the aforementioned schemes provide at least 83.8% of successful rendezvous under an indoor conference setting captured by the `Exp6` trace.

Remark 6. *The results in Fig. 3.13c indicates that the proposed PBNS-assisted schemes provide satisfactory rendezvous performance with a success rate of 80% or above.*

3.5.5 Energy Consumption

In many IoT-based DSA networks, such as for smart air quality monitoring via sensor networks [171] and mesh-based online social messaging using smart phones [2, 37], the SUs are low-complexity devices with limited power consumption. Thus, it is important to have CH schemes with low power consumption, so that the SUs can enjoy extended lifetime. In this section, we present USRP-based experiments for measuring the amount of power consumed by the proposed CH schemes.

Two X310 USRP SDRs with SBX-120 daughterboards, which provide up to $BW_{SDR} = 120$ MHz of bandwidth, are employed in our experimental test bed. Each USRP is connected to a Dell E5570 laptop running Ubuntu 16.04.2 LTS on an Intel i7-6820HQ CPU and 16GB of DDR4 memory. One USRP is used for emulating the signals transmitted from SU_a as well as the PU activities at SU_b, and the emulated signals are received by the other USRP acting as SU_b. Equivalently, the other USRP transmits SU_b's pilot signals and emulates PU activities at SU_a. Since the employed SBX-120s do not support the 5GHz U-NII bands used by LTE-U, we perform our measurements on the 3.15-3.85 GHz bands, instead. To prevent interference to the radionavigation, earth-to-satellite, and mobile activities in the 3GHz bands, we perform closed-loop simulation by connecting the USRPs via coaxial cables. Two 30dB attenuators are installed between the transmit (Tx) and receive (Rx) ports of SU_a and SU_b, as depicted in Fig. 3.14, to simulate the path loss of wireless channels. Finally, the transmit power of the SUs are set to 30dBm, while the remaining network parameters are set according to those outlined

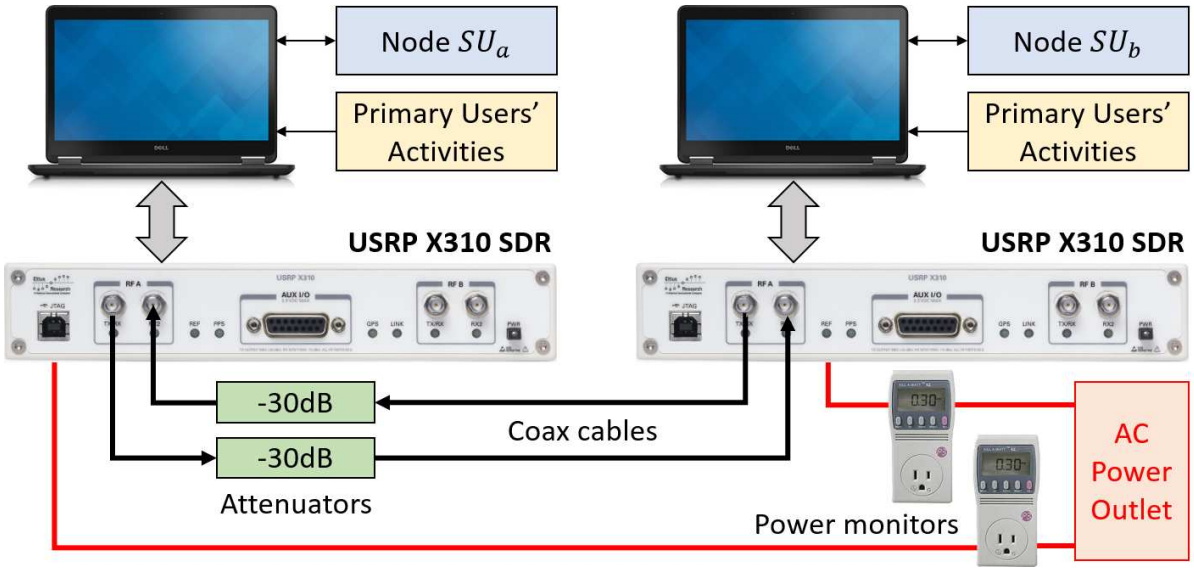


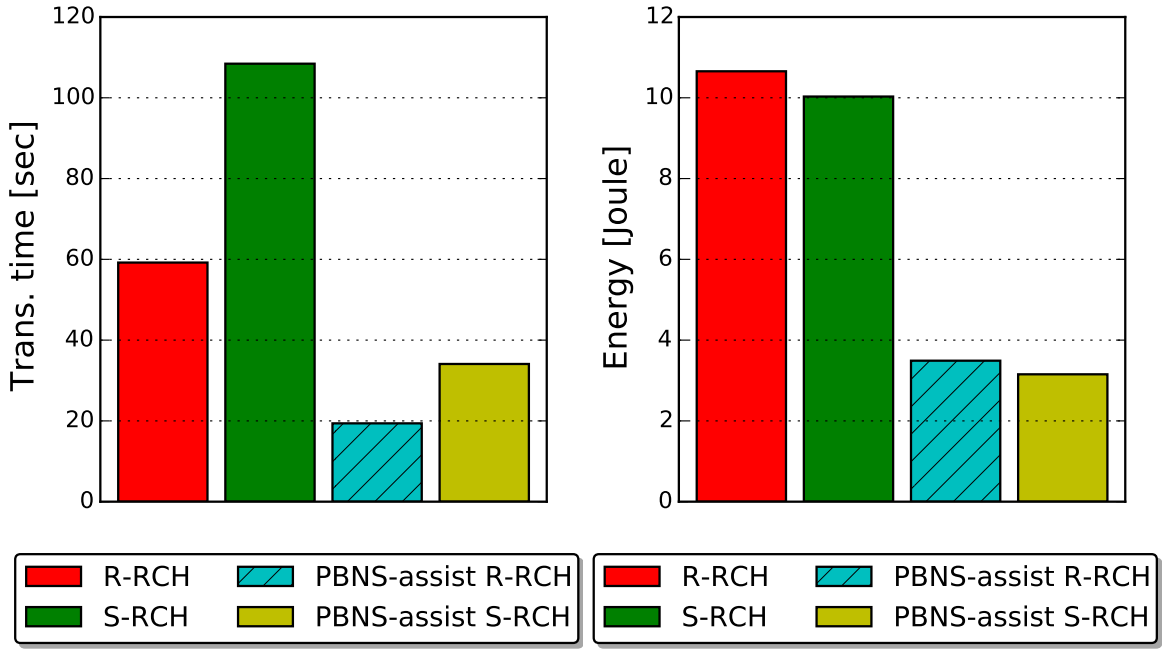
Figure 3.14: Architecture of USRP X310-based experimental test bed.

in the beginning of Section 3.5.

To proceed, we denote *transmission time* T_{tx} as the total number of times spent by nodes SU_a and SU_b for transmitting pilot signals, before rendezvous is achieved. Then, we plot the resulting transmission times of the proposed schemes in Fig. 3.15a. We observe that in the role-based (R-RCH) scheme SU_a only listens to SU_b without transmitting; thus, the transmission time of R-RCH is about half of the symmetric (S-RCH) scheme (see the red and green bars on the left). On the other hand, the PBNS-assisted schemes offer smaller expected TTRs (see the results in Fig. 3.11), while the TTR is the time spent before rendezvous and is proportional to T_{tx}. As a result, the PBNS-assisted R-RCH and S-RCH provide lower transmission times (see the cyan and yellow bars on the right).

After presenting the transmission times, we discuss how to measure SUs' power consumption, which will be used to determine the energy usage. Because signal transmission requires energy for driving the transmit amplifiers of the USRPs, a higher transmission time will cause a larger energy consumption. On the other hand, receiving signals and performing PBNS sensing also impose a non-trivial energy demand for operating the receive-side demodulation, filtering, and analog-to-digital conversion. To take into account the energy consumed by both the transmit and receive activities performed by SU nodes, we measure the power consumed by the respective USRPs using a Kill-A-Watt P4460 power monitor (see Fig. 3.14). According to the power monitors, the instantaneous power draw of SU_b (in both the R-RCH and S-RCH schemes) is equal to 32.6 Watts while that of SU_a in R-RCH is 32.2 Watts.[8]

[8]In the S-RCH scheme, SU_a consumes the same power as SU_b.



(a) Time spent transmitting signals.

(b) Total energy consumed.

Figure 3.15: Complexity and power consumption of the proposed schemes.

Notice that the instantaneous measured power represents the amount of energy (in Joules) in every second. Thus, to determine the energy consumption from measured power, we let the USRPs (SUs) to run until rendezvous, measure the expected TTR, and multiply it with the total power consumed by SU_a and SU_b. The total energy consumption is depicted in Fig. 3.15b. From the energy consumption plots, we observe that the lower transmission time of the R-RCH scheme (see the red bar) is not sufficient to reduce the total energy consumption beyond that of the S-RCH scheme (the green bar). The main reason is because the decrease in power due to the receive process (e.g., $32.6 - 32.2 = 0.4$ Watts) is not enough to overcome the increased time spent by SU_a in the receive mode, when R-RCH is employed. Such an observation also holds for the PBNS-assisted R-RCH and S-RCH schemes (see the cyan and yellow bars).

Remark 7. *The employed USRP X310 SDRs are designed for delivering high performance without trying to achieve low power consumption. As a result, a bulk of the measured power comes from the high-performance on-board digital signal processors, peripherals (Gigabit Ethernet, general purpose I/O, and JTAG programmer), and AC-to-DC electricity conversion losses. Specifically, our measurement indicates that each X310 can consume up to 31.6 Watts during idle. In mobile IoT devices, on the other hand, the idle power consumption will need to be minimized for battery lifetime maximization. As a result, most of the power (equivalently,*

energy) consumption will come from the signal transmission power, and R-RCH that has lower transmission time (see Fig. 3.15a) will consume less energy and become more beneficial.

3.6 Summary

In this chapter, we proposed a role-based (R-RCH) and symmetric (S-RCH) channel hopping schemes for rendezvous between secondary users in IoT-based dynamic spectrum access networks. Specifically, we consider the case where users have limited channel hopping (CH) range, due to their large frequency tuning time and restricted channel sensing capability. The limited CH range is taken into account by viewing the CH schemes as random walks over spectrum slice graphs, whose nodes and edges determine the set of possible channels visited at the next hop. By concentrating on the random walks, we derived the optimal transition probabilities that minimize the bounds of the maximum expected time-to-rendezvous (TTR) for both the R-RCH and S-RCH schemes. In addition to employing the optimal transition probabilities for generating CH sequences, the proposed schemes also employ partial-band Nyquist sampling (PBNS) to increase rendezvous opportunity and accelerate the TTR.

Numerical results show that the proposed PBNS-assisted R-RCH and S-RCH schemes outperform state-of-art random walk-based scheme, and achieve rendezvous with at least 80% success rate in indoor networks with realistic node mobility. Moreover, we find that the R-RCH scheme, which assigns each user with different roles prior to hopping, is more suitable for low-power IoT networks with disparate device capabilities. The S-RCH scheme, on the other hand, does not have such a requirement and may consume less energy when the users employ high-performance software-defined radios. Finally, the rendezvous opportunity—supplied by R-RCH and S-RCH—can be used to form a common channel, which is necessary for exchanging control commands prior to inter-SU communications. Since the proposed schemes account for the CH capability limits, they provide guidelines for designing communication protocols between low-complexity IoT nodes, which will be more prevalent in the future.

Chapter 4

Impacts of Dropout-Induced Contractions in D2D-Based Mobile Social Networks

4.1 Background and Motivation

Mobile social networks (MSNs) has become the most popular, go-to means of contacting relatives as well as accessing news and information, with more than one billion daily active users in 2016 alone [61]. The majority of social contents in online-based MSNs—such as Facebook, Twitter, and LinkedIn—have been exchanged using mobile device-to-cloud via cellular LTE or WiFi access points. However, the recent progress in short-range device-to-device (D2D) communications—enabled by Apple's Multipeer connectivity [9] and Android's WiFi Peer-to-Peer [13] frameworks—have motivated a decentralized way of providing MSN service on-the-go via D2D [121, 51, 8], briefed as *D2D-based MSN*. This new paradigm's ability to support social content exchange is predominantly built upon the premise that users can enjoy stable communication and networking service. In reality, however, the service provided by D2D-based MSN is far from stable thanks to hardware failure, battery depletion [170], harsh wireless medium [104], malware attack [108], and users' (in)activity [147]. These impairments render one or more users to be *dropped out* and vanish from the network without being able to send and/or receive social contents. The absence of dropped out nodes, which act as intermediary nodes to relay contents from source to destination, may diminish the D2D-based MSN's ability to provide satisfactory networking experience.

Studying the impact of user dropouts is an important primitive, especially in the current early stage of D2D-based MSN's development. At present, the prospect towards high-speed, low-power, and low-cost data transmission without being impeded by government censorship

[89] has helped real-world D2D-based MSN applications such as FireChat [89] and Jott [8] to gain popularity among millions of teenagers and students [51, 8]. It is projected that the total number of D2D-based MSN users will grow to more than 200 million by the end of 2017 [87, 116]. To be successful, D2D-based MSN must provide satisfactory networking experience to such a massive user base, which depends upon D2D-based MSN's ability to withstand initial dropouts. Understanding the impact of dropouts will not only enable predicting the success of D2D-based MSN in the future, but also shed light on the design of effective countermeasures for guaranteed networking experience.

Studying the impact of initial dropouts to the networking experience is a challenging problem due to the D2D-based MSN's structure that consists of both the communication (D2D) and social (MSN) graphs, as depicted in Fig. 4.1. In this figure, each user belonging to the social network (see the bottom graph) has a corresponding mobile device in the communication network (the upper graph). Note that there may be some D2D nodes (i.e., v_4) that act as relays with no corresponding social user. When device v_5 is dropped out from the D2D graph, the corresponding user is unable to exchange any social content and be dropped out of the MSN graph, inducing a communication-to-social (C$\rightarrow$S) dependence. As a result, user v_7 that is isolated in the MSN graph may become selfish and refuse to participate in D2D communications, which causes a social-to-communication (S$\rightarrow$C) dependence. Together, S$\rightarrow$C and C$\rightarrow$S dependencies form an *interdependence* relation, which is captured by the vertical links in Fig. 4.1. Consequently, a sequence of node isolations—due to the jointly unsuccessful communication and social contacts—continues to develop, causing a contracting (decreasing) set of residual (non-dropout) nodes referred as a *network contraction* process. If proven to scale to large, network contraction will undermine the D2D-based MSN's ability to provide satisfactory quality-of-experience (QoE), beyond that caused by the smaller initial dropouts (i.e., at node v_5 only).

Despite its potentially detrimental impact on QoE, which may invalidate D2D-based MSN as a robust alternative to conventional device-to-cloud MSN access via cellular and WiFi, little is known regarding the dropout-induced contraction process. In the social network portion of the D2D-based MSN, dropouts can be driven by users' social behavior: depending on the strength ties with their neighbors—or lack thereof—users may become inactive [147] and leave the MSN. On the other hand, the impacts of initial dropouts due to malicious software (malware) [108], random hardware breakdown [170], and jamming attacks [104] to the D2D network's reliability and connectivity [163, 29] have been widely researched. However, these researches consider initial dropouts in either MSN or D2D network only, while neglecting the inherent interdependence relation of the D2D-based MSN.

To this end, there are studies regarding networks with interdependence in the avenue of cyber-physical systems, which are mainly concerned on whether a contraction process will

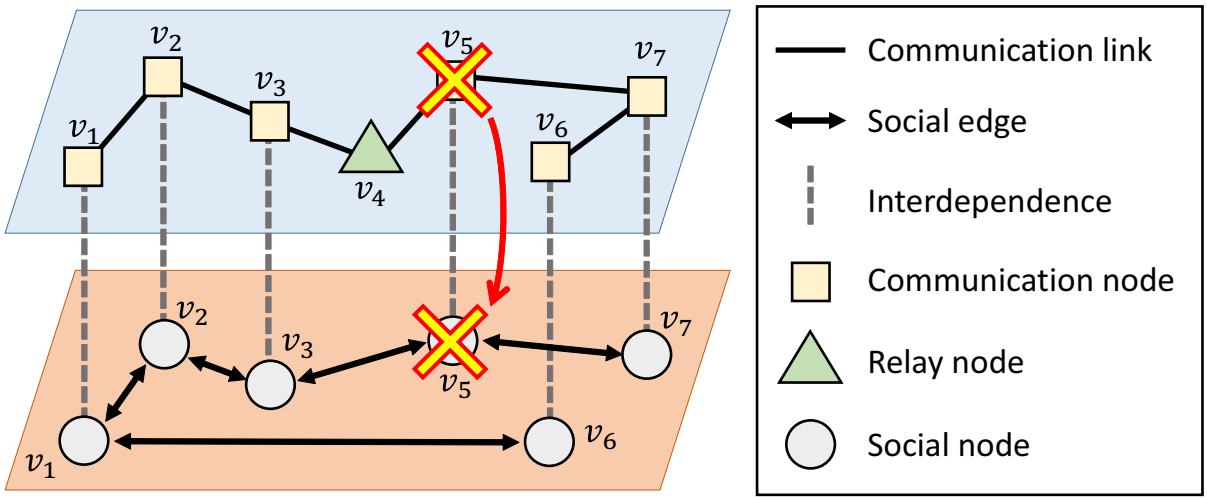


Figure 4.1: A D2D-based mobile social network (MSN).

completely disconnect the interdependent networks [38, 85]. However, even under the absence of a total disconnection, there is no guarantee that a large portion of the network will remain functional. The portion of functional nodes is proportional to how many users whose social content exchange still can be served after network contractions and is a major metric-of interest. Moreover, besides the potential impact of node dropouts to the network-wide connectivity, little is known regarding whether D2D-based MSN can provide satisfactory networking experience to its users. Thus, we are interested in analyzing the extent of the contraction process, from both the network-wide and user-centric perspectives.

To address such an open issue, we start by proposing a probabilistic model to highlight the difference between communication and social-based dropouts as the root causes of the network contraction process. Then, we employ percolation theory [38]—which describes the behavior of connected clusters in a graph after the removal of edges and nodes—to describe the temporal progress of the contraction process. To quantify the network-wide impact of initial dropouts, we define a *minimum utility ratio*, denoted as ρ_{min}, that quantifies the portion of functional nodes at the end of a contraction process, and then analyze its scaling order with respect to the (i) initial portion of residues $\bar{\rho}_0$, (ii) mean number of D2D and MSN neighbors, $\langle k \rangle_c$ and $\langle k \rangle_s$, (iii) ratio between the number of social-to-communication nodes, α, and (iv) network size N_c. Furthermore, as a user-centric QoE measure, we propose a *node resilience* Ψ_n for the likelihood that a user can enjoy a stable network connection during its ongoing social networking session. We derive the bounds of Ψ_n as well as its scaling order versus $\bar{\rho}_0$ and N_c. These analytical results are verified numerically using real-world [15] and synthetic D2D-based MSN traces, which further highlight their usefulness towards the design of countermeasure

schemes for alleviating the extent of a dropout-induced contraction process.

Our contributions can be summarized as follows.

- We introduce (α, β, c) and $(\alpha, \beta, \bar{c})$-systems to model D2D-based MSNs with (i) social-to-communication ratio α, (ii) nodes to be dropped out with likelihood proportional to the number of their neighbors raised to an exponent β, and (iii) social or communication-based dropouts with symbols c and $\bar{c}$, respectively. Using social activity trace [147], we show that $\beta = -0.132$ and highly-connected users are less likely to drop out in $(\alpha, \beta, \bar{c})$-system; otherwise, $\beta \geq 0$.

- We present a minimum utility ratio ρ_{min} and analyze its scaling order. We show that ρ_{min} increases with a complementary negative exponential (CNE)[1] factor and decreases linearly with respect to α in (α, β, c) and $(\alpha, \beta, \bar{c})$-systems, respectively. Then, the root cause of initial dropouts is determinable by checking whether ρ_{min} increases versus α.

- We further leverage the scaling order analysis that shows ρ_{min} may decrease or increase versus network size N_c, depending on the fraction of initial residues (non-dropout nodes) $\bar{\rho}_0$. Specifically, we propose a `MinUtil` admission control scheme to improve ρ_{min} by rejecting or admitting new users depending on the value of $\bar{\rho}_0$.

- We propose a node resilience metric Ψ_n and analyze its bounds to show that D2D-based MSNs with Pareto (heavy tail)-distributed content exchange session times have lower QoE than those with exponential-distributed session times. Moreover, we show via scaling order analysis that Ψ_n increases with respect to $\bar{\rho}_0$ and the network size N_c.

The rest of this chapter is organized as follows: Section 4.2 outlines the related studies in the literature. Section 4.3 describes the models of D2D-based MSN and initial dropouts. Section 4.4 formulates the network contraction process and the minimum utility ratio ρ_{min}, whose scaling order is analyzed in Section 4.5. An admission control scheme for improving ρ_{min} in Section 4.6. Section 4.7 analyzes the node resilience metric Ψ_n, while Section 4.8 concludes this chapter.

4.2 Related Works

In the literature, node dropouts in wireless communication networks have been studied extensively. For instance, battery power depletion is one of the most fundamental cause of dropouts, and many researches had focused on energy harvesting and minimization to alleviate depletion [170, 175]. Moreover, the devastating impacts of *bot-nets*—which may exploit privilege escalation to launch various attacks, including disabling the wireless interface [54]—have been studied

[1] We say that $f(x)$ scales with a complementary negative exponential factor of x if $f(x) = \Theta(-\exp\{-x\})$.

in [108]. To this end, stochastic modeling have been used to study the evolution of bot-net size [102], which can grow rapidly up to a quadratic factor over time. Besides communication-based impairments, node dropouts may instead be caused by social (MSN)-based attacks such as malware [169]-[144]—which potentially has a similar damaging impacts as bot-nets. By characterizing Sybil attacks in Renren social network, Yang *et al.* showed that existing defense mechanisms against Sybil are ineffective for MSNs [169]. The propagation of Koobface zombie in Facebook and Twitter was studied in [144], while differential equations have been proposed for modeling malware propagation in generalized social networks [48]. More recently, Viswanath *et al.* found that the rate of interaction among Facebook users may decline over time [147], thus potentially inducing dropouts. Although the aforementioned studies exemplify the vast literature regarding the causes and effects of node dropouts, most of the existing works consider either MSN or D2D network only, while neglecting the coexistence of both networks in a D2D-based MSN.

A D2D-based MSN consists of co-existing and interdependent communication and social networks. To this end, a percolation-based framework for understanding the robustness of two interdependent networks subject to uniformly-distributed initial dropouts was studied in [38]. A similar framework has been applied to cyber-physical networks in which each physical-resource node depends on multiple cyber-resource nodes [85]. Moreover, the percolation framework has been extended to include dropouts targeted toward high-degree nodes [84] and the networks with partial interdependence [57]. Note that unlike the above-mentioned studies that focus on the *percolation threshold*—the critical initial dropout ratio in which a giant connected active component starts to emerge [38, 85, 84], we are interested in the size of the largest connected component after the network contractions end, referred here as the minimum utility ratio, ρ_{min}. Specifically, we study the scaling order of ρ_{min} with respect to various network parameters in Section 4.5, to distinguish the network-wide impact of communication and social-induced initial dropouts.

In Section 4.6, we attempt to maximize ρ_{min} by proposing a `MinUtil` scheme that is largely motivated by existing admission control algorithms. The classical goal of admission control is to limit the amount of requests admitted into a particular service so that the quality-of-service of the existing users will not be degraded, while at the same time maximizing resource utilization [124]-[158]. For example, optimal schemes for admitting call requests to minimize blocking probability in cellular networks was studied in [124], while a more comprehensive survey is outlined in [19]. The admission control concept has also been adapted to IEEE 802.11 networks for maximizing throughput while maintaining fair traffic allocations [60, 68]. This concept was extended to mobile ad-hoc networks by taking into account the interference generated to neighboring nodes within the carrier-sensing range [168], for guaranteeing the service rate of primary cognitive radio users [106]. Recently, admission control is used to minimize the operating cost of

Software-as-a-Service (SaaS) cloud provider while maximizing customer satisfaction [158]. These existing studies, however, leverage admission control for solving resource allocation problems. To the best of our knowledge, little is known regarding how admission control can be used to improve the network-wide resilience and connectivity under dropouts, which is quantified in this chapter by the minimum utility ratio ρ_{min}.

The resilience of D2D network, which is an essential part of a D2D-based MSN, has been studied extensively in the literature. Given a pre-determined node density, the minimum transmission range r_0 that creates an almost surely k-connected network was derived in [29]. A fault tolerance measure to quantify the resilience of ad-hoc network's links with respect to random failures was introduced in [56]. A Semi-Markov model to characterize the stochastic impacts of node misbehavior on mobile ad-hoc network's connectivity was proposed in [162]. Moreover, the resilience of geometric random graph and the expected connection time until end-users are disconnected under lifetime-based node failures was studied in [163]. We notice that these existing studies are mainly concerned about the network-wide resilience. The resilience of end-users against initial node dropouts, however, remains a wide open problem.

In this chapter, we attempt to probe into the aforementioned open problems that are left unanswered by the existing studies. To the best of our knowledge, this chapter is the first to study D2D-based MSN by considering the communication and social networks jointly, for studying how dropout-induced network contractions impair the network-wide resilience—which is quantified by the minimum utility ratio ρ_{min} and improved with the proposed `MinUtil` admission control scheme—as well as the end-users' quality-of-experience.

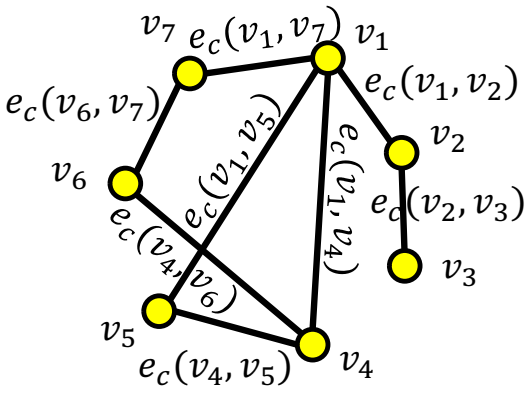
4.3 Network Model and Initial Dropouts

In this section, we describe the D2D-based MSN model. Then, we examine communication and social-based initial dropouts in D2D-based MSN.

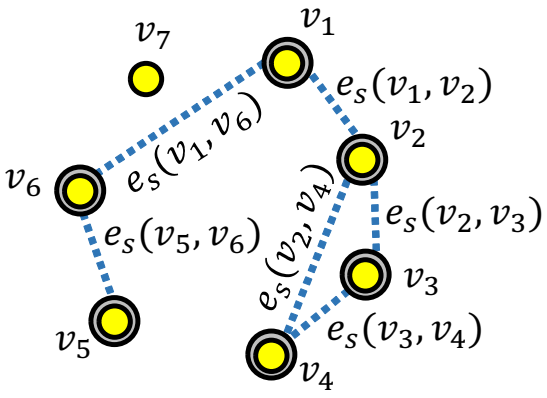
4.3.1 Network Model

Communication Graph

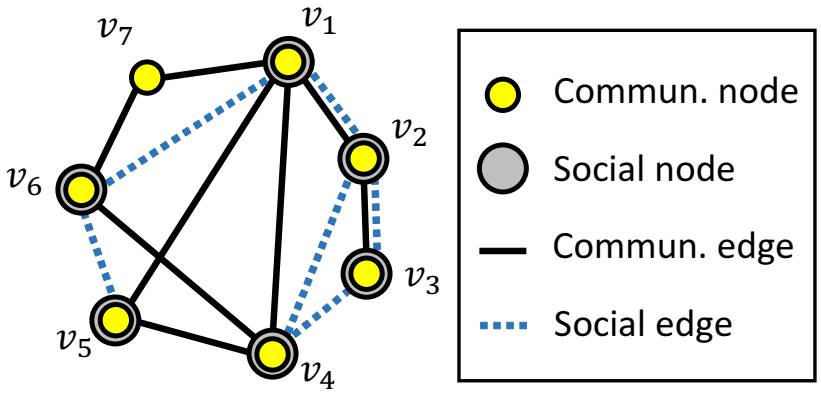
We study a network with a set of nodes (communication devices) denoted by $\mathcal{V} = \{v_1, \ldots, v_{N_c}\}$, where v_i denotes the i^{th} node and N_c is the number of nodes. Define *communication edge* as a relation that indicates two nodes can communicate directly via short-range D2D technology [9, 55]. Such an inter-node relation takes place over a discrete time span $\mathcal{T} \subset N$ called *network lifetime*, where N is the set of integers. We employ a *physical model* [79], in which data can successfully be transmitted and there is a communication edge between two nodes if the signal-to-interference ratio (SIR) exceeds a threshold [79, eq. (2)], while the SIR depends on the



(a) Communication graph $\mathcal{G}_c(\mathcal{T}_i)$.



(b) Social graph $\mathcal{G}_s(\mathcal{T}_i)$.



(c) D2D-based MSN graph $\mathcal{G}(\mathcal{T}_i)$.

Figure 4.2: Snapshot of a D2D-based MSN.

inter-node distance that is time-varying due to node mobility. Thanks to these threshold and mobility, edges may appear and disappear over time.

To proceed, we divide the lifetime $\mathcal{T}$ into disjoint time intervals $\{\mathcal{T}_i\}$, where $\mathcal{T}_i := [t_i, t_{i+1})$ denotes the i^{th} interval, $t_i \in \mathcal{T}$ is the starting time of $\mathcal{T}_i$, and $\cup_i \mathcal{T}_i = \mathcal{T}$. Similar to [105], we select the length of each interval, $t_{i+1} - t_i$, to be short so that the set of edges remain constant within this interval. In other words, there is no edge that appears nor disappears within interval $\mathcal{T}_i$. As a result, by denoting $\mathcal{E}_c(t_i)$ as the set of communication edges during the i^{th} interval, we can describe the snapshot of the D2D connection between nodes as a *communication graph* $\mathcal{G}_c(\mathcal{T}_i) := (\mathcal{V}, \mathcal{E}_c(t_i))$.

A snapshot of the communication graph, $\mathcal{G}_c(\mathcal{T}_i)$, is exemplified in Fig. 4.2a. In this figure, nodes can have edges and communicate directly only with its immediate neighbors. To enable communication between far-away nodes, multi-hop communication via store-and-forward [123],

in which data can traverse through multiple edges and nodes in order to reach their destination, is employed. For example, v_3 can send data to v_4 via a path $\{e_c(v_2, v_3), v_2, e_c(v_1, v_2), v_1, e_c(v_1, v_4)\}$. Finally, to ensure that a transmitted data can physically reach any (communication) node, we assume that $\mathcal{G}_c(\mathcal{T}_i)$ is *connected*, i.e., there is at least one path between any pair of nodes.

Social Graph

Consider the case where $N_s \leq N_c$ users utilize the nodes (i.e., smart phones and/or tablets) for exchanging social content in forms of plain text and/or multimedia files. Denote *social nodes* $\mathcal{V}_s$, where $|\mathcal{V}_s| = N_s$, as a subset of (communication) nodes $\mathcal{V}$ associated with these physical users. Moreover, let *social edge* be a relation that indicates two social nodes have a common interest and may exchange social contents with each other. By re-employing the same small time interval $\mathcal{T}_i = [t_i, t_{i+1})$ as before and denoting $\mathcal{E}_s(t_i)$ as the set of social edges within this interval, we can define a *social graph* as a tuple $\mathcal{G}_s(\mathcal{T}_i) := (\mathcal{V}_s, \mathcal{E}_s(t_i))$. This graph describes the set of users that form a network of social connections via mobile nodes, which is referred as a *mobile social network* (MSN).

A social (MSN) graph snapshot, $\mathcal{G}_s(\mathcal{T}_i)$, that corresponds to the communication graph in Fig. 4.2a is depicted in Fig. 4.2b. In this example, social edges are depicted as the dotted blue lines, while larger gray circles represent the set of social nodes $\mathcal{V}_s$. Note that although $\mathcal{V}_s \subset \mathcal{V}$, the social edges are not necessarily a subset of the communication edges in Fig. 4.2a, implying that the MSN graph snapshot $\mathcal{G}_s(\mathcal{T}_i)$ is *not* a sub-graph of $\mathcal{G}_c(\mathcal{T}_i)$. Since every newly-joining social node tends to add edges that form connected paths to existing nodes, we assume $\mathcal{G}_s(\mathcal{T}_i)$ is a connected graph.

D2D-based MSN Graph

The collection of communication edges in $\mathcal{G}_c(\mathcal{T}_i)$ can provide the necessary physical paths to deliver the social contents, whose potential endpoints are described by $\mathcal{G}_s(\mathcal{T}_i)$. For example, users $v_3, v_4 \in \mathcal{V}_s$ are a pair of endpoints and the social contents transmitted from v_3 to v_4 (see Fig. 4.2b) can physically traverse through path $\{e_c(v_2, v_3), v_2, e_c(v_1, v_2), v_1, e_c(v_1, v_4)\}$ (see Fig. 4.2a). As a result, the combination of $\mathcal{G}_c(\mathcal{T}_i)$ and $\mathcal{G}_s(\mathcal{T}_i)$ forms an MSN service over a physical D2D communication network, whose structure is jointly captured by a *D2D-based MSN graph* $\mathcal{G}(\mathcal{T}_i) := (\mathcal{G}_c(\mathcal{T}_i), \mathcal{G}_s(\mathcal{T}_i))$.

A snapshot of $\mathcal{G}(\mathcal{T}_i)$ corresponding to the communication and social graphs in Figs. 4.2a and 4.2b, respectively, is depicted in Fig. 4.2c. Note that $\mathcal{V} \setminus \mathcal{V}_s = \{v_7\}$ is a node that can help *relay* data to/from v_1 and v_6 via D2D, but cannot be an endpoint for social content exchanges. For notation simplicity in the subsequent sections, we further shift the i^{th} interval's starting

time to zero and express the shifted time interval as $\mathcal{T}_\rangle' = [0, T_i]$, where $T := t_{i+1} - t_i - 1$. As a result, by omitting the index i for simplicity, the D2D-based MSN graph can be re-stated as $\mathcal{G} = \mathcal{G}(\mathcal{T}')$.

4.3.2 Initial Dropout in D2D-based MSN

After modeling the D2D-based MSN, we describe how *dropouts*—the disconnection of some nodes from all their neighbors—may occur in such a network. In this subsection, we outline two types of dropouts as follows.

Communication-based initial dropout

In a D2D-based MSN, information is exchanged via an open wireless medium. This openness may be exploited by a vicious node to spread *malware* [108]. If multiple nodes are infected by the same malware, they can form a set of compromised nodes under the same control called *bot-net* [108]. The bot-net then attempts to expand the contagion by performing *proximity infection* [102] via communication edges. In the presence of multiple compromised nodes (bot-net), a susceptible node $v \in \mathcal{V}$ is infected with a likelihood proportional to the *communication degree*

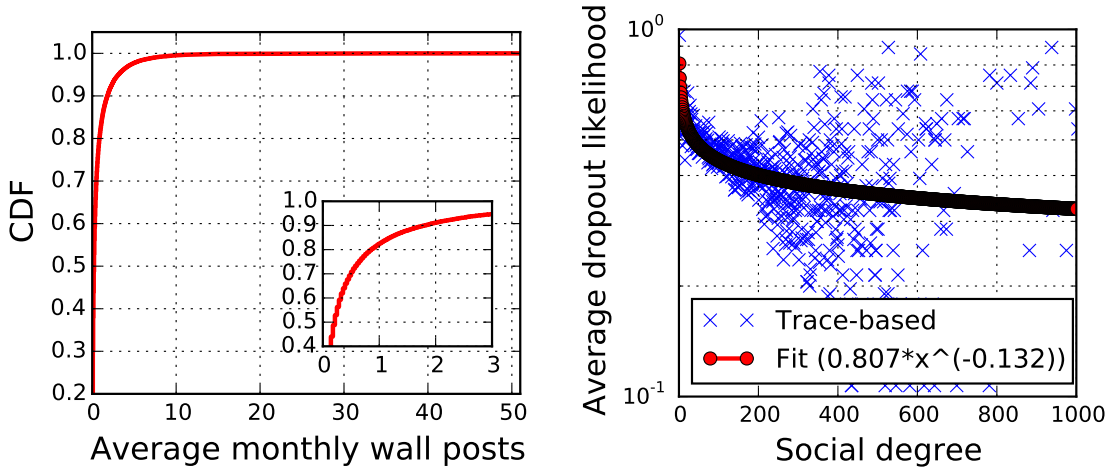
$$d_c(v) := |\{e_c(u, v) : e_c(u, v) \in \mathcal{E}_c(t_i), u \in \mathcal{V}\}|, \tag{4.1}$$

and the malware infection becomes *degree-weighted*. Further, the malware can launch a time-coordinated attack on the infected nodes. For example, it may execute any program using root permission [54], such as to disable D2D communication via the `WiFiManager` or `BluetoothAdapter` APIs [1]. As a result, compromised nodes may be dropped out due to their inability to communicate.

Alternatively, communication nodes are susceptible to faults due to hardware failure and battery power depletion [170]. These faults occur *uniformly* at random and cause nodes to become dropped out from the communication graph.

To capture these uniform and degree weighted initial node dropouts, we define the following model.

Definition 15 ((α, β, c)-system)**.** *Let initial dropout set $\mathcal{I}_0 \subset \mathcal{V}$ be the nodes that have their communication and social edges removed at time $t = 0$. Denote uniform initial dropout as the case where node $v \in \mathcal{V}$ is included into $\mathcal{I}_0$ with uniform probability, while v belongs to $\mathcal{I}_0$ with a likelihood proportional to v's communication degree in degree-weighted initial dropout. Then, (α, β, c)-system is defined as a D2D-based MSN in which nodes suffer a uniform or degree-weighted initial dropouts when $\beta = 0$ or $\beta = 1$, respectively.*



(a) CDF of average monthly posts.

(b) Likelihood of dropout.

Figure 4.3: Analysis of New Orleans regional Facebook network [147].

Social-based initial dropout

In a D2D-based MSN, the mobile social network (MSN) serves as a medium for users with social relation (friendship) to exchange information. The performance of such an exchange is mainly driven by the users as the agents and, in particular, their social activity. To better understand users' activity, we examine the Facebook online social network, in which a friend can post comments to a user's *wall*, and then these comments can then be seen by others visiting the user's profile. *Wall posts* represent a broadcast-type messaging service and is one of the most popular methods of user interaction [155].

We employ the New Orleans regional data set [147] that collects the wall post history between 90,269 Facebook users from September 2006 to January 2009. We use the time stamps of wall posts to examine each user's level of activity. In particular, we assume that a user is *active* if it generates at least one wall post in that month; otherwise, it is *non-active*. Fig. 4.3a shows the CDF of the average monthly wall posts. Observe that less than 17.5% of users (see the inset) are *highly-active* (i.e., have more than one average posts per month). Although a minority of the total population, these users generate the majority of social networking traffic and will mostly enjoy the reduced data roaming fee [4] and lower energy consumption benefits offered by D2D-based MSN. Thus, we consider these highly-active users as the social nodes $\mathcal{V}_s$.

From the availability perspective, each of the social nodes may become non-active and unreachable during certain months, thus resembling a dropout node. Let the *dropout likelihood* of a user $v \in \mathcal{V}_s$ be the number of month it becomes non-active over the total number of months

(i.e., 28 in the New Orleans data set), and the *social degree*

$$d_s(v) := |\{e_s(u,v) : e_s(u,v) \in \mathcal{E}_s(t_i), u \in \mathcal{V}_s\}| \tag{4.2}$$

be the number of social neighbors connected to v. We are interested in the relation between the average dropout likelihood versus the social degree, as depicted in Fig. 4.3b. After applying a least-square (LS) fit to the scatter plot, we observe that user v drops out with a likelihood of $0.807 \times d_s(v)^{-0.132}$, which means users with fewer social neighbors are dropped out more often. Accordingly, we present the following definition.

Definition 16 ($(\alpha, \beta, \bar{c})$-system)**.** *Let $\alpha := \frac{N_s}{N_c}$ be a social-to-communication ratio. Denote initial dropout set $\mathcal{I}_0 \subset \mathcal{V}_s$ as the social nodes whose edges are removed at time $t = 0$. Node $v \in \mathcal{V}_s$ is included into $\mathcal{I}_0$ with probability proportional to $d_s(v)^\beta$, with $\beta = -0.132$. Then, $(\alpha, \beta, \bar{c})$-system represents a D2D-based MSN with social-based initial node dropouts.*

According to Definition 16, there are more communication nodes that also act as social nodes as α increases. Moreover, symbol $\bar{c}$ is used to emphasize that the initial dropout is social-based (not communication-induced).

Remark 8. *Definitions 15 and 16 indicates a clear distinction between social and communication-based initial dropouts. In the former, $\beta \geq 0$ and the nodes are dropped out uniformly at random or with probability proportional to the communication degree. In the latter, $\beta = -0.132$ (i.e., $\beta < 0$) and nodes with higher social degrees are dropped out less often.*

4.4 Network Contraction Process and Problem Formulation

Next, we study how the dissimilar $(\alpha, \beta, \bar{c})$ and (α, β, c)-systems will trigger different network contraction processes. Then, we define a minimum utility ratio metric and state our objective of analyzing the scaling order of this metric.

4.4.1 Network Contraction Process

Due to the disconnection of the initial dropout nodes' edges at time $t = 0$, several residual nodes may become isolated from others and become non-active. This "isolate-and-deactivate" process can be repeated by more and more nodes. Therefore, a cascade of node dropouts that rapidly *contracts* (shrinks) the number of active nodes in the network may develop from scratch, given sufficient time. Such an initial dropout-induced *network contraction* could be very detrimental to mobile nodes as well as the network's operation.

In order to understand the potential worst-case impact of initial dropouts, we need to investigate how network contraction evolves over time. This is studied via a process defined as follows.

Definition 17 (Network contraction process)**.** *Let $\mathcal{R}_c(t) \subset \mathcal{V}$ and $\mathcal{R}_s(t) \subset \mathcal{V}_s$ be the set of residual communication and social nodes at time $t \geq 1$, with initial values $\mathcal{R}_c(1) = \mathcal{V} \setminus \mathcal{I}_0$ and $\mathcal{R}_s(1) = \mathcal{V}_s \setminus \mathcal{I}_0$. Let $\mathcal{A}_c(t) \subset \mathcal{R}_c(t)$ and $\mathcal{A}_s(t) \subset \mathcal{R}_s(t)$ be the set of active communication and social nodes at time t, respectively. Then, a network contraction process is defined as a decreasing sequence of the sets of active nodes, which is denoted as a bi-variate process $\{\mathcal{A}_c(t), \mathcal{A}_s(t)\}_{t\geq 1}$.*

In Definition 17, residual nodes are the set of non-dropouts, while active nodes are the subsets of the residues that can still function properly in the D2D-based MSN. To give a more detailed illustration regarding the residual and active nodes, consider the network contraction processes for the $(\alpha, \beta, \bar{c})$ and (α, β, c)-systems as follows.

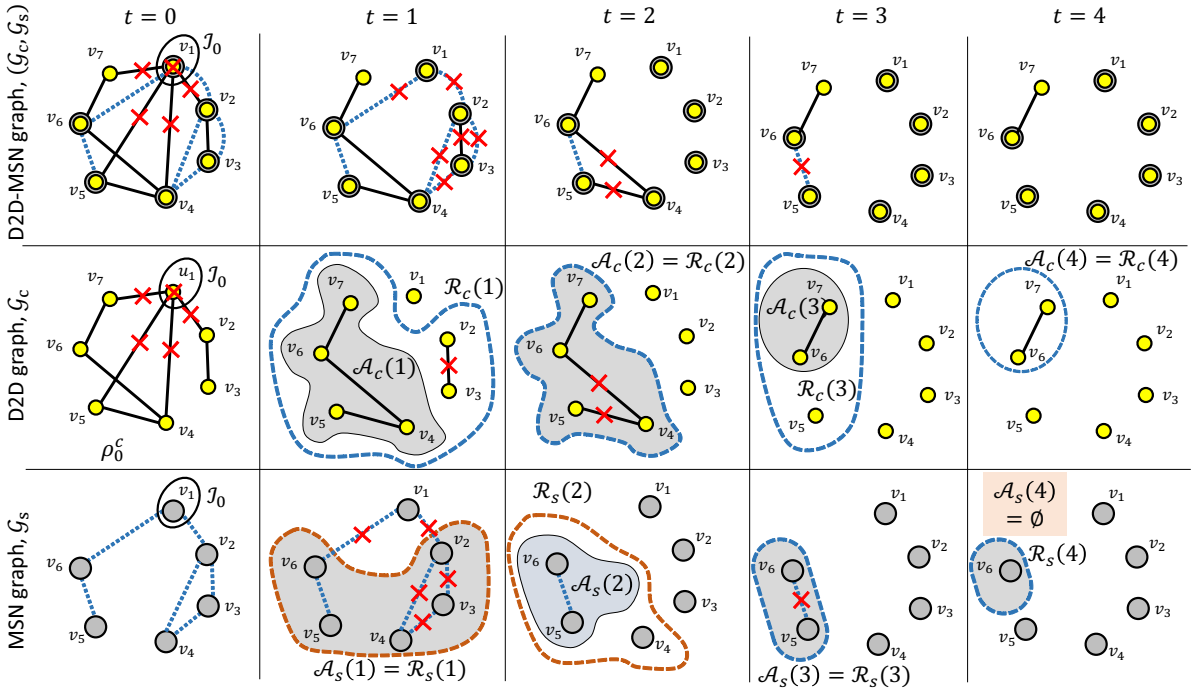


Figure 4.4: Illustration of a communication dropout-induced network contraction process.

Communication-based initial dropout

We examine an (α, β, c)-system where $\beta = 1$ and node $\mathcal{I}_0 = \{v_1\}$ that has the largest communication degree is dropped out at time $t = 0$. At the next step, the residual communication nodes $\mathcal{R}_c(1)$ (see Fig. 4.4 at $t = 1$) is divided into two connected components, so that only $|\mathcal{A}_c(1)| = |\{v_4, \ldots, v_7\}| = 4$ nodes belong to the largest connected component (LCC) and remain active.

To proceed, we outline how social node dropouts will occur at time $t = 2$. Based on our D2D-based MSN model in Section 4.3.1, in order to exchange social contents with its social neighbors, a node must be connected to (i.e., be *active* in) the communication graph. Conversely, the social users $\{v_2, v_3\}$ corresponding to the previously non-active communication nodes cannot exchange social contents and must be dropped out of the social graph. We refer such a relationship as a *communication-to-social* (C→S) dependence. This dependence leaves $\mathcal{R}_s(2) = \{v_4, v_5, v_6\}$ as the set of residual social nodes, which are divided into two connected components (see the area with dotted red boundary at time $t = 2$), and node v_4 that does not belong to the largest component becomes non-active. Consequently, only $|\mathcal{A}_s(2)| = |\{v_5, v_6\}| = 2$ social nodes remain active. The number of active communication nodes, on the other hand, remains equal to its previous value, that is, $|\mathcal{A}_c(2)| = |\mathcal{A}_c(1)|$.

Next, we explain how the set of non-active social nodes can induce communication node dropouts at time $t = 3$. To this end, users tend to share more contents if they see more contributions from their friends (social neighbors) [40], while the amount of contents is proportional to their social degrees [27, Fig. 9],[147, Fig. 2]. Conversely, non-active users that are isolated and cannot see any contribution from their friends will use the MSN less often and eventually close the D2D-based MSN app on their devices. To conserve battery life, the devices corresponding to these users will have their D2D communication features disabled and be dropped out. We refer such a communication dropout due to social non-activity as a *social-to-communication* (S→C) dependence.

Owing to the S→C dependence, node v_4 becomes dropped out at time $t = 3$. This results in the set of $\mathcal{R}_c(3) = \{v_5, v_6, v_7\}$ residues, while only $|\mathcal{A}_c(3)| = |\{v_6, v_7\}| = 2$ communication nodes remain active. The number of active social nodes remain unchanged (e.g., $|\mathcal{A}_s(3)| = |\mathcal{A}_s(2)|$).

By advancing one step further and re-invoking the C→S dependence, observe that the contraction process finally *ends* at step $t = 4$.[2] Here, the number of active social nodes is $|\mathcal{A}_s(4)| = 0$ because the set of residual social nodes is completely divided into singletons, with no connected component. For simplicity, we assume that the time until a network contraction process ends (i.e., $t = 4$ in this example) is smaller than the D2D-based MSN graph's stationary interval $T_i = t_{i+1} - t_i - 1$ (see Section 4.3.1).

[2]We say a network contraction process has ended at time t if $|\mathcal{A}_c(t)| = |\mathcal{A}_c(t+1)|$ and $|\mathcal{A}_s(t)| = |\mathcal{A}_s(t+1)|$.

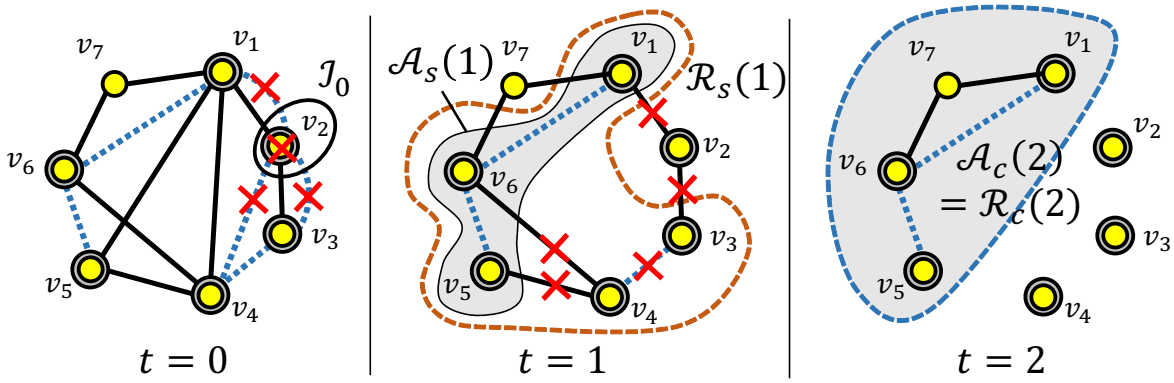


Figure 4.5: Illustration of a social dropout-induced contraction process.

Social-based initial dropout

We consider an $(\alpha, \beta, \bar{c})$-system where user $\mathcal{I}_0 = \{v_2\}$ is initially dropped out and has its social edges removed (see Fig. 4.5 for $t = 0$). In the next step, $t = 1$, the set of residual social nodes $\mathcal{R}_s(1)$, which is indicated as the area with red-dotted boundary in Fig. 4.5, is divided into two connected components. To examine the worst-case scenario, only nodes $\mathcal{A}_s(1) = \{v_1, v_5, v_6\}$ belonging to the LCC of $\mathcal{G}_s$ remain active.

Thanks to the S→C dependence, nodes $\{v_2, v_3, v_4\}$ corresponding to the previously non-active users are also dropped out from $\mathcal{G}_c$. The residual nodes $\mathcal{R}_c(2)$, indicated as the region with dotted boundary in Fig. 4.5 for $t = 2$, now all belong to the LCC and the network contraction process *stops* with $|\mathcal{A}_c(2)| = |\{v_1, v_5, v_6, v_7\}| = 4$ active nodes.

4.4.2 Problem Formulation

By comparing Figs. 4.4 and 4.5, we notice that the social and communication-based initial dropouts may produce different amounts of active nodes. We will leverage this difference to distinguish these two dropouts, by analyzing their worst-case impact to the network contraction.

For $(\alpha, \beta, \bar{c})$-system, the number of active social and communication nodes are changed only at odd and even steps, respectively. To simplify analysis for large N_s and N_c, the number of active nodes are converted to their *fractions*, respectively denoted as $\rho_s(t) := \frac{|\mathcal{A}_s(t)|}{N_s} \in [0, 1]$ and $\rho_c(t) := \frac{|\mathcal{A}_c(t)|}{N_c} \in [0, 1]$. The contraction process can now be represented as $\{\rho(t)\}_{t \geq 1}$, where $\rho(t)$ is the *utility ratio* at time t, which is equal to $\rho_c(t)$ if t is odd and to $\rho_s(t)$ for even t. Note that the term "utility" is used since $\rho(t)$ represents the fraction of nodes/users that can still *utilize* (i.e., communicate using) the D2D-based MSN.

On the other hand, the network contraction process for (α, β, c)-system can be defined similarly, but with $\rho(t) = \rho_c(t)$ for odd t and $\rho(t) = \rho_s(t)$, otherwise.

Since network contraction results from both the S→C and C→S dependencies, our interest is on the even time steps when both dependencies have taken place. Notice that $\rho(t)$ is a random variable that depends on the set of initial dropouts $\mathcal{I}_0$, an outcome of a probabilistic selection process. Moreover, $\rho(t)$ is a decreasing sequence. Thus, we define the following metric to quantify the worst-case yet random impact of initial dropouts.

Definition 18. *Let $\bar{\rho}(t) := E(\rho(t))$ be the mean utility ratio at time t. Then, minimum utility ratio is defined as*

$$\rho_{min} := \inf_t \{\bar{\rho}(t) : t \text{ is even}\} = \lim_{k\to\infty} \bar{\rho}(2k). \tag{4.3}$$

Before using this ratio for problem formulation, let x represent each of the following parameters.

1. **Average social and communication degrees**, $\langle k\rangle_s$ and $\langle k\rangle_c$, quantifying how connected the D2D-based MSN is.
2. **Mean fraction of initial residues**, denoted respectively as $\bar{\rho}_0 := 1 - E[\frac{|\mathcal{I}_0|}{N_s}]$ and $\bar{\rho}_0 := 1 - E[\frac{|\mathcal{I}_0|}{N_c}]$ for $(\alpha, \beta, \bar{c})$ and (α, β, c)-systems, indicating severity of initial dropouts.
3. **Social-to-communication ratio** $\alpha = \frac{N_s}{N_c}$, measuring how many communication nodes have S→C dependence.

To proceed, we define the following concept of scaling order.

Definition 19 (Scaling order)**.** *Let $\rho_{min}(x)$ be the maximum isolation time as a function of x. Then, (i) $\rho_{min}(x) = O(f(x))$ if there exist constants $c, x_0 > 0$ such that $\rho_{min}(x) \leq cf(x)$ for all $x \geq x_0$, (ii) $\rho_{min}(x) = \Omega(f(x))$ means $f(x) = O(\rho_{min}(x))$, while (iii) $\rho_{min}(x) = \Theta(f(x))$ if $\rho_{min}(x) = O(f(x))$ and $\rho_{min}(x) = \Omega(f(x))$.*

Given Definitions 18 and 19, our main objective is to analyze the scaling order of the minimum utility ratio for $(\alpha, \beta, \bar{c})$ and (α, β, c)-systems. Specifically, we ask, *"what are the distinguishing scaling order characteristics of ρ_{min} between $(\alpha, \beta, \bar{c})$ and (α, β, c)-systems?"*

4.5 Scaling Order Analysis

We have defined our main objective of studying the distinct properties of $(\alpha, \beta, \bar{c})$ and (α, β, c)-systems via scaling order analysis. In this section, we reach this objective by describing $\bar{\rho}(t)$ as a time-recursive sequence to derive the self-consistent equations of the minimum utility ratio, ρ_{min}. Then, we leverage such equations to analyze the scaling order of ρ_{min} with respect to various network parameters.

4.5.1 Time-Recursive Mean Utility Ratio

Before analyzing the minimum utility ratio, we describe the temporal progress of the mean utility ratio $\bar{\rho}(t)$. To do this, we first consider an (α, β, c)-system and start from the odd steps. Let $\bar{\chi}_c(t) := E[\frac{|\mathcal{R}_c(t)|}{N_c}]$ be the mean fraction of social residues and $\psi_s(\bar{\chi}_c(t)) := \frac{\bar{\rho}(t)}{\bar{\chi}_c(t)}$ be the portion of *active* residues at odd t. For the even steps, let $\bar{\chi}_s(t) := E[\frac{|\mathcal{R}_s(t)|}{N_s}]$ be the mean fraction of social residues, while $\psi_s(\bar{\chi}_s(t)) := \frac{\bar{\rho}(t)}{\bar{\chi}_s(t)}$ is the portion of active residues. Then, we have the result outlined in Lemma 9.

Note that analyzing the progress of $\bar{\rho}(t)$ boils down to the problem of finding the portion of residual nodes in each time step via *percolation*—the process of removing edges and nodes from the network. However, unlike existing results for general interdependent and smart grid networks [38, 85], we assume not all communication nodes have one-to-one interdependence to social nodes, which is captured by the social-to-communication ratio of $\alpha \leq 1$. Thus, the existing results [38, 85] cannot be used here directly and we employ the step-by-step derivation below, instead.

Lemma 9. *Let $\bar{\rho}_0 := E\frac{|\mathcal{V}\setminus\mathcal{I}_0|}{N_c}$ be the mean fraction of initial residues. In (α, β, c)-systems, the mean utility ratio for even t is equal to $\bar{\rho}(t) = \bar{\chi}_s(t) \times \psi_s(\bar{\chi}_s(t))$, where $\bar{\chi}_s(t)$ satisfies*

$$\begin{cases} \bar{\chi}_s(t) = \bar{\rho}_0 \times \psi_c(\bar{\chi}_c(t-1)), \\ \bar{\chi}_c(t) = \bar{\rho}_0[1 - \alpha[1 - \psi_s(\bar{\chi}_s(t-1))]]. \end{cases} \tag{4.4}$$

Proof. The progress of mean utility ratio, $\{\bar{\rho}(t) : t \geq 1\}$, is traced by employing the following step-by-step approach.

Step 1 ($t = 1$): Initially, the mean fraction of communication residues is equal to $\bar{\chi}_c(1) = \bar{\rho}_0$. Then, the mean utility ratio is given as $\bar{\rho}(1) = \bar{\chi}_c(1)\psi_c(\bar{\chi}_c(1)) = \bar{\rho}_0\psi_c(\bar{\rho}_0)$.

Step 2 ($t = 2$): Every social user has a reciprocal communication node (see Fig.4.2c). Consequently, users corresponding to the non-active nodes are dropped out due to C→S dependence. The fraction of social residues becomes equal to the mean utility ratio at the previous time step, i.e.,

$$\bar{\chi}_s(2) = \bar{\rho}(1) = \bar{\rho}_0\psi_c(\bar{\rho}_0), \tag{4.5}$$

while the mean utility ratio is given as $\bar{\rho}(2) = \bar{\chi}_s(2)\psi_s(\bar{\chi}_s(2))$.

Step 3 ($t = 3$): The communication nodes corresponding to the previously non-active social users are also dropped out due to S→C dependence. This represents an additional $\alpha[1 - \psi_s(\bar{\chi}_s(2))]$ fraction of dropped out communication nodes, since only an α fraction of nodes also

act as social users. The fraction of residual communication nodes becomes

$$\bar{\chi}_c(3) = \bar{\chi}_c(1)[1 - \alpha[1 - \psi_s(\bar{\chi}_s(2))]], \tag{4.6}$$

while the mean utility ratio is equal to $\bar{\rho}(3) = \bar{\chi}_c(3)\psi_c(\bar{\chi}_c(3))$.

Step 4 ($t = 4$): Owing to C→S dependence, a fraction $1 - \frac{\bar{\rho}(3)}{\bar{\chi}_c(3)}$ of communication nodes had become non-functional at the previous step. Similarly, among the subset of communication nodes that have one-to-one interdependence with the social users, a $1 - \frac{\bar{\rho}(3)}{\bar{\chi}_c(3)}$ fraction of them will become dropped out, causing the same amount of non-functional social users. As a result, the fraction of residual (non-dropout) social nodes becomes

$$\bar{\chi}_s(4) = \bar{\rho}(2) - \left(1 - \frac{\bar{\rho}(3)}{\bar{\chi}_c(3)}\right)\bar{\rho}(2), \tag{4.7}$$

and we can show that $\bar{\chi}_s(4) = \frac{\psi_c(\bar{\chi}_c(3))}{\psi_c(\bar{\chi}_c(1))}\bar{\rho}(2)$ by following the procedure in [85, eq. (13)]. This means, from $\bar{\rho}(2)$ to $\bar{\chi}_s(4)$, an additional fraction of $1 - \frac{\psi_c(\bar{\chi}_c(3))}{\psi_c(\bar{\chi}_c(1))}$ social nodes have to be dropped out. Thus, the total fraction of dropouts from $t = 0$ to $t = 4$ becomes

$$\begin{aligned}(1 - \bar{\chi}_s(2)) + \bar{\chi}_s(2)\left(1 - \frac{\psi_c(\bar{\chi}_c(3))}{\psi_c(\bar{\chi}_c(1))}\right) &= 1 - \frac{\bar{\chi}_s(2)}{\psi_c(\bar{\chi}_c(1))}\psi_c(\bar{\chi}_c(3)) \\ &= 1 - \bar{\rho}_0\psi_c(\bar{\chi}_c(3)),\end{aligned} \tag{4.8}$$

where the second line holds because $\bar{\chi}_s(2)$ satisfies (4.5). As a result, the fraction of social residues becomes equal to

$$\bar{\chi}_s(4) = \bar{\rho}_0\psi_c(\bar{\chi}_c(3)), \tag{4.9}$$

and the mean utility ratio is given as $\bar{\rho}(2) = \bar{\chi}_s(2)\psi_s(\bar{\chi}_s(2))$.

General time steps: By expanding to larger time steps, we find that the mean utility ratio $\bar{\rho}(t)$ and the fraction of residual nodes, $\bar{\chi}_c(t)$ and $\bar{\chi}_s(t)$, satisfy Lemma 9. □

For $(\alpha, \beta, \bar{c})$-system, $\bar{\chi}_c(t)$ and $\psi_c(\bar{\chi}_c(t)))$ are defined only for even time steps, while the social parameters, $\bar{\chi}_s(t)$ and $\psi_s(\bar{\chi}_s(t))$, are for odd steps. Then, the result below can be obtained by following the proof of Lemma 9.

Lemma 10. *For $(\alpha, \beta, \bar{c})$-system, let $\bar{\rho}_0 := E\frac{|\mathcal{V}_s \setminus \mathcal{I}_0|}{N_s}$. Then, the mean utility ratio equals $\bar{\rho}(t) = \bar{\chi}_c(t) \times \psi_c(\bar{\chi}_c(t))$ for even t, where $\bar{\chi}_c(t)$ satisfies*

$$\begin{cases}\bar{\chi}_c(t) = 1 - \alpha[1 - \bar{\rho}_0\psi_s(\bar{\chi}_s(t-1))], \\ \bar{\chi}_s(t) = \bar{\rho}_0 \times \psi_c(\bar{\chi}_c(t-1)).\end{cases} \tag{4.10}$$

Next, we analyze the mean portions of active residues, ψ_c and ψ_s. To do this, let $P_c(k) := \Pr\{d_c(u) = k : u \in \mathcal{V}\}$ and $P_s(k) := \Pr\{d_s(v) = k : v \in \mathcal{V}_s\}$ be the *communication* and *social degree distributions*, respectively.

Lemma 11. *The portion of active social and communication residues are respectively given as*

$$\psi_s(\bar{\chi}_s(t)) = 1 - G_{s,0}(1 - (1 - f_s)\bar{\chi}_s(t)), \tag{4.11}$$

$$\psi_c(\bar{\chi}_c(t)) = 1 - G_{c,0}(1 - (1 - f_c)\bar{\chi}_c(t)), \tag{4.12}$$

where f_s and $G_{s,0}$ are both functions of the social degree distribution $P_s(k)$ and the initial dropout exponent β. On the other hand, f_c and $G_{c,0}$ are functions of β and the communication degree distribution $P_c(k)$.

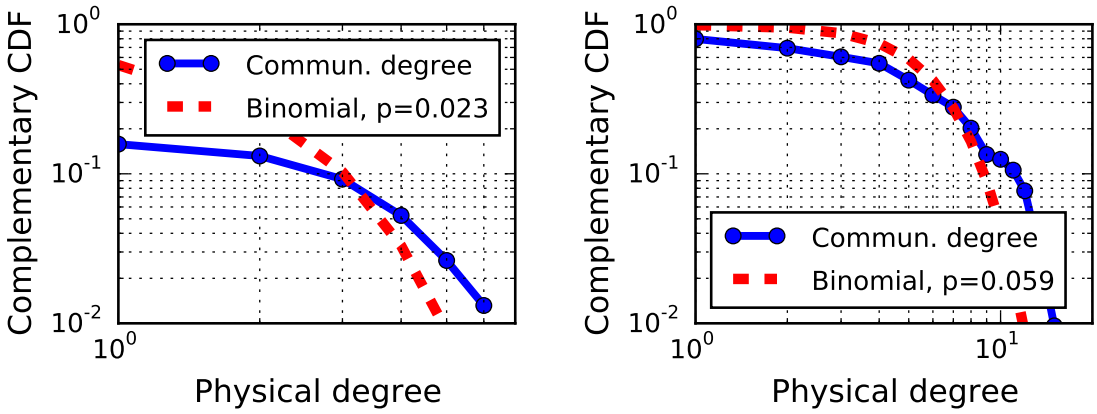
Proof. See [38, pp. 1027] for the proof of the $(\alpha, 0, c)$-system, and [84] for the $(\alpha, 1, c)$- and $(\alpha, \beta, \bar{c})$-systems. □

4.5.2 Minimum Utility Ratio in Traces

The minimum utility ratio is a function of the mean utility ratio $\bar{\rho}(t)$, which according to Lemmas 9-11 depends on the communication and social degree distributions that can be studied using mobile network traces. Hence, we use mobile network traces to analyze the minimum utility ratio.

Mobile network traces collect users' access to base stations or access points (e.g., infrastructure-based traces), locations (e.g., GPS-based traces), or Bluetooth contacts over time (e.g., direct contact-based traces). Because D2D-based MSN exploits short-range D2D contacts for social content delivery, we choose the direct contact-based traces. Moreover, since D2D-based MSN is useful for users sharing a common social interest, mobile network traces with social groups are preferred. Therefore, we select `sigcomm2009` [15], `realitymining` [14], and `socialblueconn` [16] data sets that include the traces of Bluetooth sightings and friendship relation between nodes in classroom and conference settings.

In `sigcomm2009` trace, 100 smart phones are distributed to participants of the ACM SIGCOMM 2009 conference from August 17th to 19th, 2009. Numbers of contacts, contact times, and Facebook friendship among nodes are collected. Data from 76 smart phones are used in analysis, while that from others are discarded due to inactivity. On the other hand, 100 Nokia 6600 phones are handed out to the faculty and students of MIT's Media Laboratory and Sloan School of Business from July 2004 to May 2005 in the `realitymining` trace. In this trace, the number of contacts and contact times between nodes are recorded. Finally, the `socialblueconn` trace collects the Facebook friendship of 15 students at the University of Calabria, Italy from January 21 to February 2nd, 2014.



(a) sigcomm2009, $t_i = 1,500$ min. (b) realitymining, $t_i = 100$ hours.

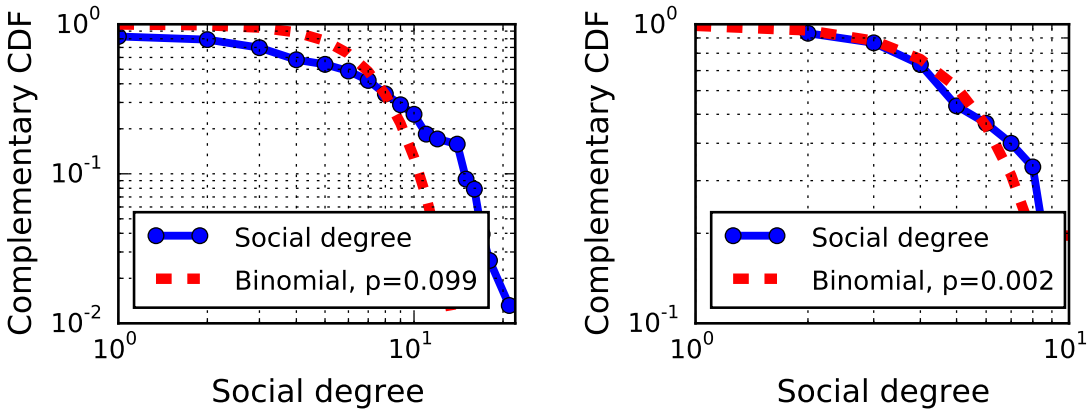
Figure 4.6: Communication degree distribution of real traces.

We proceed by examining the communication and social degrees of the aforementioned traces, which according to Lemma 9-11 are needed to analyze the minimum utility ratio. We start by selecting an arbitrary time t_i, collecting the number of neighbors from the communication graph snapshot at time t_i, and calculating the empirical communication degree distribution, $P_c(k)$, of the `sigcomm2009` and `socialblueconn` traces. From the log of the empirical complementary cumulative distribution function (CCDF) versus the log of communication degree plots in Figs. 4.6a and 4.6b that are *non-linear*, we find that the degree distribution $P_c(k)$ does not follow a power law. In contrast, the Binomial CCDF fits the resulting empirical $P_c(k)$ very well, as indicated by the small mean-squared errors (MSEs) of 7.14% and 6.67%, respectively. Thus, we capture the communication graph $\mathcal{G}_c$ using an Erdös-Rényi (ER) model [59] that is known to have a Binomial degree distribution.

We also examine the empirical social degree distribution, $P_s(k)$, of the `sigcomm2009` and `socialblueconn` traces in Figs. 4.7a and 4.7b. As before, we assume that the social graph $\mathcal{G}_s$ can be represented using ER model, since the least-square (LS) fit using Binomial CCDF produces small MSEs of 5.5% and 7.8%, respectively.

Finally, we concentrate on the Binomial degree fittings for the `sigcomm2009` trace in Figs. 4.6a and 4.7a. We observe that the mean communication degree, $\langle k \rangle_c = \sum k P_c(k) \approx 1.725$, is *smaller* than the mean social degree, $\langle k \rangle_s = \sum k P_s(k) \approx 6.825$, which indicates one-hop, direct transmission is not sufficient to support social content exchange. This supports our assumption in Section 4.3.1 that multi-hop communication is needed for data transmission.

After knowing that communication and social degrees are Binomial-distributed, we are ready to analyze the minimum utility ratio, ρ_{min}, which is equivalent to finding the solution



(a) sigcomm2009 trace [15]. (b) socialblueconn trace [16].

Figure 4.7: Social degree distribution of real traces.

to the self-consistent equation when the time recursive equation in (4.4) or (4.10) converges. This has been solved in [85, eq. (21)] for interdependent networks with *symmetric* graphs ($\langle k\rangle_c = \langle k\rangle_s$) and a *perfect* one-to-one interdependence ($\alpha = 1$). However, we assume a less connected communication graph snapshot ($\langle k\rangle_c < \langle k\rangle_s$) and a more generic $\alpha \in (0, 1]$, such that the one-step analysis for obtaining [85, eq. (21)] from the time-recursive equation cannot be adopted directly to our case. Instead, we consider a substitution-based approach, which leads to the *self-consistent* equations[3] in Proposition 2 below.

To obtain the results in Proposition 2, we assume that the number of nodes is arbitrarily large (i.e., $N_c \uparrow \infty$). The case of finite N_c, on the other hand, will be considered in Section 4.6.

Proposition 2. *In D2D-based MSNs with arbitrarily large number of nodes, the minimum utility ratio satisfy*

$$\rho_{min} = \begin{cases} \bar{\rho}_0(1-\exp\{-\langle k\rangle_c w_c^2 \alpha \rho_{min}\})(1-\exp\{-\langle k\rangle_s \rho_{min}\}), & \text{for } (\alpha,\beta,c)\text{-system,} \\ \left(1-\alpha(1-\bar{\rho}_0)-\alpha\bar{\rho}_0\exp(-\langle k\rangle_s w_s^2(\rho_{min}+\alpha-1))\right)(1-\exp(-\langle k\rangle_c\rho_{min}))\,, & \\ & \text{for } (\alpha,\beta,\bar{c})\text{-system,} \end{cases}$$

where $w_s = G_s^{-1}(\bar{\rho}_0)$ *for* $(\alpha,\beta,\bar{c})$*-system is a solution to* $G_s(x) = \sum_{k=0}^{\infty} P_s(k)x^{k^{-0.132}}$*, while* $w_c = \frac{\log(\bar{\rho}_0)}{\langle k\rangle_c} + 1$ *and* $w_c = 1$ *for* $(\alpha, 1, c)$ *and* $(\alpha, 0, c)$*-systems, respectively.*

Proof. To analyze the minimum utility ratio, we consider the following disjoint cases.

Minimum utility ratio for $(\alpha, 1, c)$-system. According to Definition 18, ρ_{min} is the

[3]An equation of the minimum partition ratio is self-consistent if it is in form of $\rho_{min} = f(\rho_{min})$, where $f : R \to R$ is a real function.

smallest possible $\bar{\rho}(t)$ for even t, a function of a time-decreasing $\bar{\mathcal{X}}_s(t)$. Thus, the minimum utility is equal to $\rho_{min} = \bar{\chi}_s \psi(\bar{\chi}_s)$, after setting the fractions of social residues in (4.4) to $\bar{\chi}_s(t) = \bar{\chi}_s(t+2) = \bar{\chi}_s$.

To proceed, consider the portion of active residues, ψ_c, as a function of the communication degree distribution $P_c(k)$. Recall that according to the `sigcomm2009` trace [15], $P_c(k)$ is Binomially-distributed, which converges to a Poisson random variable for large network. By denoting $\bar{\mathcal{X}}_c := \bar{\mathcal{X}}_c(t)$, setting $\beta_c = 1$, and plugging $P_c(k)$ into [84, eqs. (6)-(11)], we obtain $G_{c,0}(\bar{\chi}_c) = \exp\{\langle k \rangle_c w_c^2 (\bar{\chi}_c - 1)\}$ and

$$f_c = \exp\{-\langle k \rangle_c w_c^2 \bar{\chi}_c (1 - f_c)\}, \tag{4.13}$$

where $w_c = G_c^{-1}(\bar{\rho}_0) = \frac{\log(\bar{\rho}_0)}{\langle k \rangle_c} + 1$. This can further be substituted into (4.11) to get

$$\psi_c(\bar{\chi}_c) = 1 - f_c. \tag{4.14}$$

The social degree can also be approximated into a Poisson distribution. Plugging this into [84, eqs. (6)-(11)] obtains

$$G_{s,0}(\bar{\chi}_s) = G_{s,1}(\bar{\chi}_s) = \exp\{\langle k \rangle_s (\bar{\chi}_s - 1)\}, \tag{4.15}$$

$$f_s = \exp\{-\langle k \rangle_s \bar{\chi}_s (1 - f_s)\}, \text{ and} \tag{4.16}$$

$$\psi_s(\bar{\chi}_s) = 1 - f_s. \tag{4.17}$$

Finally, by combining (4.14), (4.17), and Lemma 9, the minimum utility ratio can be analyzed as follows.

$$\rho_{min} = (\bar{\rho}_0(1 - f_c))(1 - f_s) \tag{4.18}$$

$$= \bar{\rho}_0 \left(1 - e^{-\langle k \rangle_c w_c^2 \bar{\chi}_c (1 - f_c)}\right) \left(1 - e^{-\langle k \rangle_s \bar{\chi}_s (1 - f_s)}\right) \tag{4.19}$$

$$= \bar{\rho}_0 \left(1 - e^{-\langle k \rangle_c w_c^2 \bar{\chi}_c \psi_c(\bar{\chi}_c)}\right) \left(1 - e^{-\langle k \rangle_s \bar{\rho}_0 (1 - f_c)(1 - f_s)}\right) \tag{4.20}$$

$$= \bar{\rho}_0 \left(1 - e^{-\langle k \rangle_c w_c^2 \alpha \rho_{min}}\right) \left(1 - e^{-\langle k \rangle_s \rho_{min}}\right). \tag{4.21}$$

The right side of (4.18) is obtained by plugging the first line of (4.4) and (4.17) into $\bar{\chi}_s \psi_s(\bar{\chi}_s)$. Eq. (4.19) holds by plugging (4.13) and (4.16) to the line above it, while (4.20) can be obtained because $\bar{\chi}_s = \bar{\rho}_0 \psi(\bar{\chi}_c) = \bar{\rho}_0 (1 - f_c)$, according to (4.14) and the first line of (4.4). To get (4.21), notice that $\bar{\chi}_c \psi_c(\bar{\chi}_c)$ is proportional to the cardinality of set $\mathcal{A}'_c(\infty)$, the communication nodes that also act as social nodes and remain active after all nodes have been dropped out, over N_c. Since $\alpha = \frac{N_s}{N_c}$ and the ratio $\frac{1}{N_s} |\mathcal{A}'_c(\infty)|$ is equal to ρ_{min}, we obtain $\bar{\chi}_c \psi_c(\bar{\chi}_c) = \alpha \rho_{min}$. Combining this and (4.18) into (4.20) results in the second line of (4.13) for $\beta = 1$.

Minimum utility ratio for $(\alpha, 0, c)$-system. According to [84], this is the case when $w_c = 1$. Plugging this to the steps above obtains the second line of (4.13) for $\beta = 0$.

Minimum utility ratio for $(\alpha, \beta, \bar{c})$-system. In this case, we find the smallest fraction of communication residues by setting $\bar{\chi}_c(t) = \bar{\chi}_c(t+2) = \bar{\chi}_c$. By employing Poisson-distributed $P_c(k)$ and $P_s(k)$ into [84, eqs. (6)-(11)] and setting $\beta = -0.132$, we obtain $G_{c,0}(\bar{\chi}_c) = \exp\{\langle k\rangle_c(\bar{\chi}_c - 1)\}$, $f_c = \exp\{-\langle k\rangle_c \bar{\chi}_c(1-f_c)\}$, $G_{s,0}(\bar{\chi}_s) = \exp\{\langle k\rangle_s w_s^2(\bar{\chi}_s - 1)\}$, and $f_s = \exp\{-\langle k\rangle_s w_s^2 \bar{\chi}_s(1-f_s)\}$, where $w_s = G_s^{-1}(\bar{\rho}_0)$ and $G_s(x) = \sum_{k=0}^{\infty} P_s(k) x^{k^{-0.132}}$ while the fractions of active residues are equal to (4.14) and (4.17). With these variables, we can solve $\rho_{min} = \bar{\chi}_c \times \psi_c(\bar{\chi}_c)$ according to Lemma 10 by following the steps in (4.18)-(4.20), to get the first line of (4.13). □

The minimum utility ratio ρ_{min} can be calculated from (4.13) by performing a binary search over a continuous interval $[0, 1]$, as outlined in [39]. By letting

$$\rho_{cr} := \min\{\bar{\rho}_0 : \rho_{min} > \epsilon\} \tag{4.22}$$

be a *critical* initial dropout ratio [84] and $\epsilon > 0$ be a small constant, there are two possible solutions: (i) $\rho_{min} \leq \epsilon$ (i.e., near zero) in the *subcritical* regime ($\bar{\rho}_0 < \rho_{cr}$), while (ii) $\rho_{min} > \epsilon$ when $\bar{\rho}_0 \geq \rho_{cr}$ (i.e., $\bar{\rho}_0$ is *supercritical*). We can show that $\epsilon = 0$ for (α, β, c)-system and $\epsilon < (1-\alpha)$ for $(\alpha, \beta, \bar{c})$-system by numerically solving (4.13). In both cases, no user can exchange social contents via the D2D-based MSN.[4] Thus, we are interested in the supercritical case in which there are users that still can access the network.

4.5.3 Scaling Order of Minimum Utility Ratio

After outlining that our main interest is on the supercritical case ($\bar{\rho} \geq \rho_{cr}$), we are now ready to analyze the scaling orders of minimum utility ratio. We start by examining the scaling order with respect to the average communication degree $\langle k\rangle_c$ as follows.

Theorem 6. *For $\bar{\rho}_0 \geq \rho_{cr}$, the scaling order of ρ_{min} with respect to the average communication degree $\langle k\rangle_c$ satisfies*

$$\rho_{min} = \begin{cases} \Theta(-\exp(-b_1[(\log(\bar{\rho}_0))^2 \langle k\rangle_c^{-1} + \langle k\rangle_c])), & \text{for } (\alpha, \beta, c)\text{-system}, \\ \Theta(-\exp(-\langle k\rangle_c \rho^*_{min})), & \text{for } (\alpha, \beta, \bar{c})\text{-system}, \end{cases} \tag{4.23}$$

where $b_1 = \alpha\rho^*_{min}$ is a constant and ρ^*_{min} is the smallest ρ_{min} such that the left and right sides of (4.13) intersect non-trivially.

[4] In $(\alpha, \beta, \bar{c})$-system, although $\rho_{min} > 0$ when $\bar{\rho}_0 < \rho_{cr}$ (i.e., initial dropouts are subcritical), all the remaining nodes are relays only.

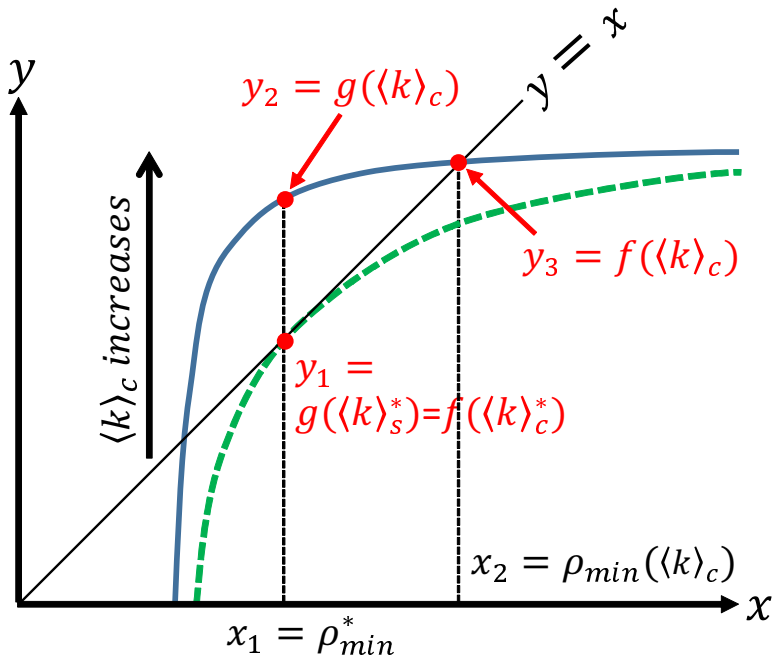


Figure 4.8: Illustration for proof of Theorem 6.

Proof. Unlike similar scaling order problems where the variable-of-interest (i.e., $\bar{\rho}_0$ in our case) can be expressed in *closed-form* [79], we only have the *self-consistent* equation (4.13) at hand. Since conventional approaches cannot be applied, we combine Definition 19 with a graphical analysis, which is detailed as follows.

Scaling order of (α, β, c)-system. The solution of (4.13) is the largest crossing point between plots $y = \rho_{min}$ and

$$y = \bar{\rho}_0(1 - e^{-\langle k\rangle_c w_c^2 \alpha \rho_{min}})(1 - e^{-\langle k\rangle_s \rho_{min}}), \tag{4.24}$$

where $x = \rho_{min}$ and y represents the horizontal and vertical axes of the plots. Consider the case where the average communication degree $\langle k\rangle_c$ increases, while variables $\bar{\rho}_0$, α, and $\langle k\rangle_s$ are fixed, as depicted in Fig. 4.8. Let $\langle k\rangle_c^*$ be the smallest $\langle k\rangle_c$ such that the minimum utility ratio is non-trivial ($\rho_{min} > \epsilon$), and ρ_{min}^* be the minimum utility ratio corresponding to $\langle k\rangle_c^*$. Moreover, denote $f(\langle k\rangle_c)$ as the right side of (4.24) and let $g(\langle k\rangle_c) := (1 - e^{-\langle k\rangle_c w_c^2 \alpha \rho_{min}^*}) \times \bar{\rho}_0(1 - e^{-\langle k\rangle_s \rho_{min}^*})$, where the last terms are constant since ρ_{min}^* is fixed. We want to prove the scaling order in the first line of (4.23) by employing Definition 19 and showing that $f(\langle k\rangle_c) = \Theta(g(\langle k\rangle_c))$. To this end, we observe the followings.

1. By comparing points y_2 and y_3 in Fig. 4.8, it is easy to see that $f(\langle k\rangle_c) \geq g(\langle k\rangle_c)$ for any $\langle k\rangle_c \geq \langle k\rangle_c^*$.

2. Let $c_2 := \frac{1}{g(\langle k\rangle_c^*)}$. By comparing points y_1 and y_2 in Fig. 4.8, observe that $g(\langle k\rangle_c) \geq g(\langle k\rangle_c^*)$ for all $\langle k\rangle_c \geq \langle k\rangle_c^*$ such that $c_2 g(\langle k\rangle_c) \geq c_2 g(\langle k\rangle_c^*) = 1$. On the other hand, by the definition of minimum utility ratio, $f(\langle k\rangle_c) = \rho_{min} \leq 1$. Based on these observations, we conclude $\exists c_2 > 0$ such that $\forall \langle k\rangle_c \geq \langle k\rangle_c^*, f(\langle k\rangle_c) \leq c_2 g(\langle k\rangle_c)$.

After combining these results and by the fact that $\rho_{min} = f(\langle k\rangle_s)$, we can show that ρ_{min} scales with the same order as $g(\langle k\rangle_c)$. In other words, $\rho_{min} = \Theta(g(\langle k\rangle_c)) = \Theta(\bar{\rho}_0(1-e^{-\langle k\rangle_c w_c^2 \alpha \rho_{min}^*})(1-e^{-\langle k\rangle_s \rho_{min}^*}))$. Recall that $\bar{\rho}_0$, α, $\langle k\rangle_s$, and ρ_{min}^* are fixed such that the term $\bar{\rho}_0(1-\exp\{-\langle k\rangle_s \rho_{min}^*\})$ is constant. Dropping this term obtains the second line of (4.23).

Scaling order of $(\alpha, \beta, \bar{c})$-system. By employing the second line of (4.13), re-applying the graphical approach in Fig. 4.8, and following the steps above, we can show that $\rho_{min} = \Theta\big((1-\exp\{-\langle k\rangle_c \rho_{min}^*\}) \times (1-\alpha(1-\bar{\rho}_0) - \alpha\bar{\rho}_0 \exp(-\langle k\rangle_s w_s^2(\rho_{min}+\alpha-1)))\big)$. After dropping the second multiplication term, we finally obtain the first line of (4.23). □

By increasing the mean social degree $\langle k\rangle_s$, fixing other variables ($\bar{\rho}_0$, α, and $\langle k\rangle_c$), and following the proof of Theorem 6, we get the following scaling order versus $\langle k\rangle_s$ in Corollary 6. Note that the result for $(\alpha, \beta, \bar{c})$-system is missing because the first line of (4.13) is a function of $w_s = G_s^{-1}(\bar{\rho}_0)$, which requires solving $G_s(x) = \sum_{k=0}^{\infty} P_s(k) x^{k^{-0.132}}$. To this end, $G_s(x)$ is a diverging sequence due to the exponent $k^{-0.132}$ and is hard to be analyzed in closed-form.

Corollary 6. *For (α, β, c)-system with $\bar{\rho}_0 \geq \rho_{cr}$, the scaling order of the minimum utility ratio versus the average social degree $\langle k\rangle_s$ satisfies*

$$\rho_{min} = \Theta(-\exp(-\langle k\rangle_s \rho_{min}^*)). \tag{4.25}$$

On the other hand, the scaling order versus the fraction of initial residues, $\bar{\rho}_0$, satisfies the following corollary.

Corollary 7. *Let $b_2 = \exp(-b_1\langle k\rangle_c)$. When $\bar{\rho}_0 > \rho_{cr}$, the scaling order of ρ_{min} with respect to the fraction of initial residues $\bar{\rho}_0$ for (α, β, c)-system satisfies*

$$\rho_{min} = \Theta\left(\bar{\rho}_0\left(1 - b_2 \exp\left\{-b_1[(\log(\bar{\rho}_0))^2 \langle k\rangle_c^{-1} + 2\log(\bar{\rho}_0)]\right\}\right)\right).$$

Proof. The scaling order for (α, β, c)-system can be proved by varying α, fixing $\bar{\rho}_0$, $\langle k\rangle_c$, and $\langle k\rangle_s$, and then re-applying the steps in the proof of Theorem 6. The proof for $(\alpha, \beta, \bar{c})$-system, on the other hand, is outlined below.

Let $f(\alpha)$ be the right-hand side of the self-consistent equation. As illustrated in Fig. 4.9, $f(\alpha)$ becomes lower as α is increased. Denote ρ_{min}^* as the minimum utility ratio corresponding

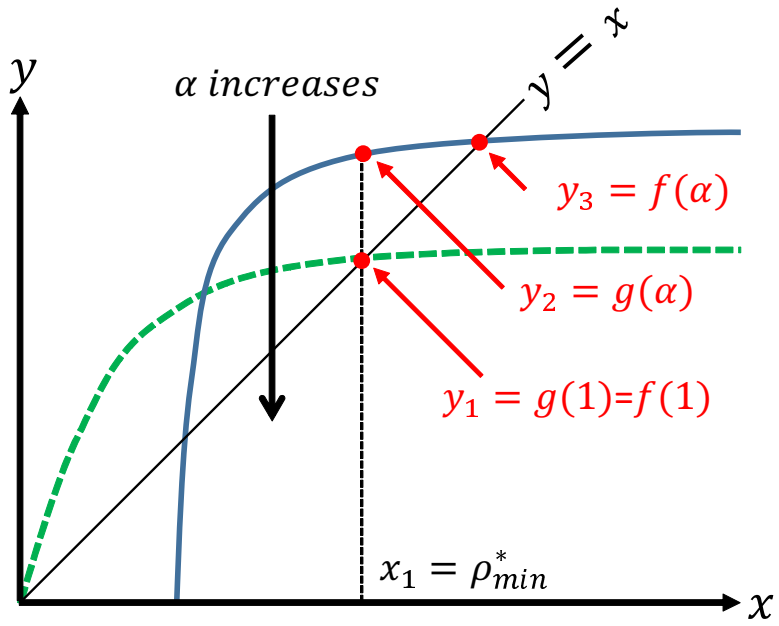


Figure 4.9: Illustration for proof of Corollary 8.

to $\alpha = 1$ and let $g(\alpha) = f(\alpha)|_{\rho_{min}:=\rho^*_{min}}$ such that $g(1) = f(1)$. *Firstly*, we compare $y_2 = g(\alpha)$ to $y_3 = f(\alpha)$ in Fig. 4.9 and observe that $\forall \alpha \in [0, 1], f(\alpha) \geq g(\alpha)$. *Secondly*, after re-visiting the figure and comparing y_2 to $y_1 = g(1)$, we notice $\forall \alpha \in [0, 1], g(\alpha) \geq g(1)$ such that $c_2 g(\alpha) \geq 1$ holds by setting $c_2 = \frac{1}{g(1)}$. Combining this and the definition of minimum utility ratio (i.e., $\rho_{min} = f(\alpha) \leq 1$) obtains $f(\alpha) \leq 1 \leq c_2 g(\alpha)$ for all $0 \leq \alpha \leq 1$.

Put together, these two results indicate that $\rho_{min} = f(\alpha) = \Theta\big(\big(1 - e^{-\langle k\rangle_c \rho_{min}}\big)(1 - \alpha(1 - \bar{\rho}_0) - \alpha b_4 e^{-\alpha\langle k\rangle_s w_s^2})\big)$, where $b_4 := \bar{\rho}_0 \exp\{-(1 - \rho^*_{min})\langle k\rangle_s w_s^2\}$. Notice that the constants "1" and "$1 - \exp\{-\langle k\rangle_c \rho^*_{min}\}$" are insignificant and can be removed. Finally, by dropping "$-\alpha b_4 \exp\{-\alpha\langle k\rangle_s w_s^2\}$" that grows slower than the dominant linear term "$\alpha(1 - \bar{\rho}_0)$," we obtain the first line of (4.26). □

As before, the scaling order analysis versus $\bar{\rho}_0$ for $(\alpha, \beta, \bar{c})$-system is missing because $w_s = G_s^{-1}(\bar{\rho}_0)$ is hard to derive. In the following subsection, we will numerically show that the scaling orders in (4.25)-(4.26) also hold for $(\alpha, \beta, \bar{c})$-system.

Finally, the scaling order of ρ_{min} versus the social-to-communication ratio α follows the corollary below.

Corollary 8. *For $\bar{\rho}_0 \geq \rho_{cr}$, the scaling order of ρ_{min} with respect to the social-to-communication*

ratio α satisfies

$$\rho_{min} = \begin{cases} \Theta(-\alpha), & \textit{for } (\alpha, \beta, \bar{c})\textit{-system}, \\ \Theta(-\exp(-\alpha b_3)), & \textit{for } (\alpha, \beta, c)\textit{-system}, \end{cases} \tag{4.26}$$

where $b_3 = \langle k \rangle_c w_c^2$ is a constant.

Proof. The scaling order for (α, β, c)-system can be proved by varying α, fixing $\bar{\rho}_0$, $\langle k \rangle_c$, and $\langle k \rangle_s$, and then re-applying the steps in the proof of Theorem 6. The proof for $(\alpha, \beta, \bar{c})$-system, on the other hand, is outlined below.

Let $f(\alpha)$ be the right-hand side of the self-consistent equation. As illustrated in Fig. 4.9, $f(\alpha)$ becomes lower as α is increased. Denote ρ^*_{min} as the minimum utility ratio corresponding to $\alpha = 1$ and let $g(\alpha) = f(\alpha)|_{\rho_{min} := \rho^*_{min}}$ such that $g(1) = f(1)$. *Firstly*, we compare $y_2 = g(\alpha)$ to $y_3 = f(\alpha)$ in Fig. 4.9 and observe that $\forall \alpha \in [0, 1], f(\alpha) \geq g(\alpha)$. *Secondly*, after re-visiting the figure and comparing y_2 to $y_1 = g(1)$, we notice $\forall \alpha \in [0, 1], g(\alpha) \geq g(1)$ such that $c_2 g(\alpha) \geq 1$ holds by setting $c_2 = \frac{1}{g(1)}$. Combining this and the definition of minimum utility ratio (i.e., $\rho_{min} = f(\alpha) \leq 1$) obtains $f(\alpha) \leq 1 \leq c_2 g(\alpha)$ for all $0 \leq \alpha \leq 1$.

Put together, these two results indicate that $\rho_{min} = f(\alpha) = \Theta\big(\big(1 - e^{-\langle k \rangle_c \rho_{min}}\big)(1 - \alpha(1 - \bar{\rho}_0) - \alpha b_4 e^{-\alpha \langle k \rangle_s w_s^2})\big)$, where $b_4 := \bar{\rho}_0 \exp\{-(1 - \rho^*_{min}) \langle k \rangle_s w_s^2\}$. Notice that the constants "1" and "$1 - \exp\{-\langle k \rangle_c \rho^*_{min}\}$" are insignificant and can be removed. Finally, by dropping "$-\alpha b_4 \exp\{-\alpha \langle k \rangle_s w_s^2\}$" that grows slower than the dominant linear term "$\alpha(1 - \bar{\rho}_0)$," we obtain the first line of (4.26). □

Theorem 6 indicates that the minimum utility ratio grows with a CNE factor^2 of the mean communication degree $\langle k \rangle_c$. Corollary 8, on the other hand, suggests a dichotomy: (i) ρ_{min} *decreases* linearly versus α in $(\alpha, \beta, \bar{c})$-system, while (ii) it *increases* with a CNE factor^2 of the social-to-communication ratio α in (α, β, c)-system. This means, the scaling order of ρ_{min} versus α is a *distinguishing characteristic* between social and communication-induced network partition processes.

4.5.4 Numerical Evaluation

In this section, we validate our analytical findings via numerical experiments. Unless stated otherwise, we employ a social-to-communication ratio of $\alpha = 0.8$, generate 50 D2D-based MSN graph realizations, and select 100 different initial dropouts in each realization. All experiments are performed using `NetworkX` framework in `Python`.

Validity of Erdös-Rényi Graph Modeling

We start by verifying our assumption that the structure of real-world D2D-based MSN can well be represented as synthetic ER graphs. Specifically, we want to show that realistic and synthetic

networks exhibit comparable minimum utility ratios. To do this, we employ the `sigcomm2009` trace [15] and take the snapshots of $\mathcal{G}_c$ and $\mathcal{G}_s$ at a measurement time $t_m = 1,500$ minutes to represent the *realistic* network. For the *synthetic* network, we employ two ER graphs with mean communication and social degrees of $\langle k \rangle_c = 1.725$ and $\langle k \rangle_s = 6.825$ to match the `sigcomm2009` trace's snapshots. The minimum utility ratio ρ_{min} of $(\alpha, 1, c)$ and $(\alpha, \beta, \bar{c})$-systems employing the realistic and synthetic networks are plotted versus the fraction of initial dropouts $\bar{\rho}_0$ in Fig. 4.10. Although there exist small gaps for intermediate values of $\bar{\rho}_0$ (see the plots with same colors in Fig. 4.10), both the realistic and synthetic traces produce similar curve shapes, indicating an *identical scaling order*.

Validity of Proposition 2

Next, we verify the self-consistent equations of ρ_{min} in (4.13), which is derived by assuming a large network size. To do so, we need to apply an increasing number of nodes, N_c, which cannot be done with *fixed* measurement-based trace [15]. Thus, we employ synthetic ER graphs for the social and communication graphs, $\mathcal{G}_s$ and $\mathcal{G}_c$. We vary N_c, while the number of social nodes follows $N_s = \alpha N_c$. Moreover, we employ the mean degrees of $\langle k \rangle_c = 6$ and $\langle k \rangle_s = 10$ to ensure connected $\mathcal{G}_c$ and $\mathcal{G}_s$.[5] The resulting ρ_{min} for the $(\alpha, 0, c)$-system is plotted in Fig. 4.11a. As N_c increases, the numerical results converge to the theoretical solution of (4.13), thus validating Proposition 2. Although not presented here, similar result holds for $(\alpha, \beta, \bar{c})$ and $(\alpha, 1, c)$-systems.

Remark 9. *The gap between the theoretical and numerical results for the subcritical regime ($\bar{\rho}_0 < \rho_{cr} = 0.33$) in Fig. 4.11 suggests that the result in Proposition 4.11a may not be exact when N_c is small. We will analyze the impact of such a small and finite N_c to the minimum utility ratio in Section 4.6 below.*

Several distinguishing properties between the $(\alpha, \beta, \bar{c})$, $(\alpha, 0, c)$, and $(\alpha, 1, c)$-systems can be observed by setting the number of communication nodes to $N_c = 100$ and plotting the resulting ρ_{min} in Fig. 4.11b. Firstly, these three systems have distinct critical initial dropout ratios of ρ_{cr} of 0.20, 0.33, and 0.41, respectively. Secondly, the $(\alpha, 1, c)$-system has the lowest ρ_{min}, while the $(\alpha, \beta, \bar{c})$-system exhibits the best (highest) performance. This is because initial dropouts in the former are concentrated on highly-connected nodes, which are important hubs for content exchange. In the latter, initial dropouts only occur on the social users, which is a subset of the total nodes and have a more limited impact.

Remark 10. *According to Fig. 4.11b, Proposition 2 can be used to distinguish network contractions due to communication-induced ($(\alpha, 0, c)$ and $(\alpha, 1, c)$-systems) and social-based dropouts*

[5] An ER graph with N nodes is almost surely connected if the mean degree $\langle k \rangle$ satisfies $\langle k \rangle > (1-\epsilon)\frac{N-1}{N}\ln(N)$ for a small $\epsilon > 0$.

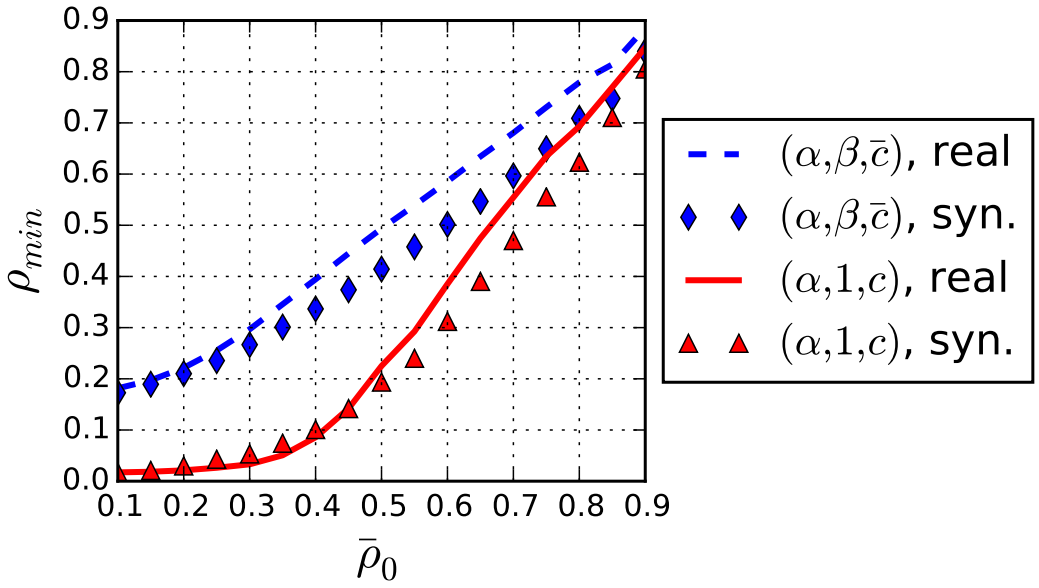


Figure 4.10: Minimum utility ratio of real-life [15] and synthetic traces.

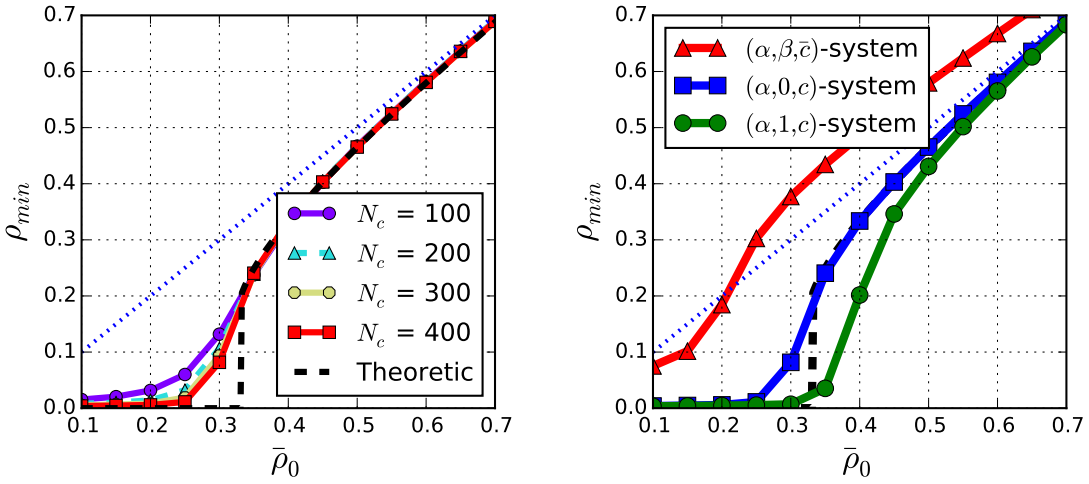
($(\alpha,\beta,\bar{c})$-system). To do this, we estimate $\bar{\rho}_0$ and let $\rho_{min}^{(th)}$ be the minimum utility ratio of the $(\alpha,0,c)$-system calculated from (4.13). Then, we determine the D2D-based MSN is an $(\alpha,\beta,\bar{c})$, $(\alpha,0,c)$, or $(\alpha,1,c)$-system if the measured ρ_{min} is larger than, equal to, or smaller than $\rho_{min}^{(th)}$.

Remark 11. *To get $\bar{\rho}_0$ and ρ_{min} in practical D2D-based MSNs like FireChat [89] and Jott [37], the developer and/or operator can poll a subset of nodes called sensors. Then, $\bar{\rho}_0$ and ρ_{min} can be estimated from the portions of active sensors at origin ($t = 0$) and at the end of network contractions (i.e., after the number stabilizes), respectively.*

Scaling Order of Minimum Utility Ratio

In this subsection, we first examine the scaling order of ρ_{min} versus the mean communication degree, $\langle k\rangle_c$. To do this, we re-use the network parameters in Fig. 4.11 and set $\bar{\rho}_0 = 0.85$, which ensures a supercritical regime ($\bar{\rho}_0 \geq \rho_{cr}$) and a non-trivial ρ_{min}. Fig. 4.12a plots the minimum utility ratio ρ_{min} as $\langle k\rangle_c$ is increased. We observe that for the $(\alpha,1,c)$, $(\alpha,0,c)$, and $(\alpha,\beta,\bar{c})$-systems, ρ_{min} increase with a CNE factor[2] of $\langle k\rangle_c$, thus verifying the analysis in Theorem 6. Such an increase is possible because when $\langle k\rangle_c$ is high, there are more alternative paths to maintain a large connected component and many functional nodes after a dropout.

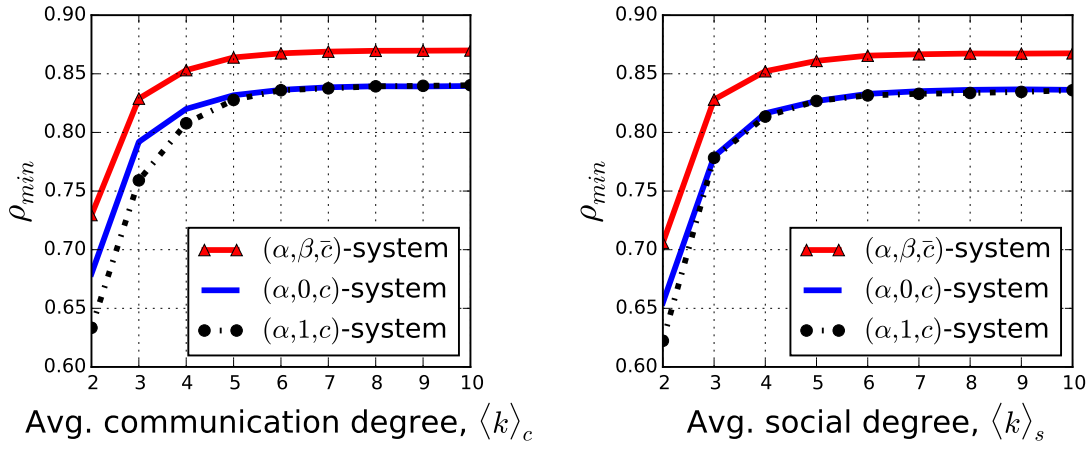
A similar CNE increase with respect to $\langle k\rangle_s$ can also be observed for both the $(\alpha,1,c)$ and $(\alpha,0,c)$-systems in Fig. 4.12b, verifying our analysis in Corollary 6. From this figure, we find that the minimum utility ratio for $(\alpha,\beta,\bar{c})$-system follows the same CNE scaling order.



(a) $(\alpha, 0, c)$-system.

(b) Various initial dropouts.

Figure 4.11: Scaling order of minimum utility ratio versus initial dropout ratio.



(a) Mean communication degree.

(b) Mean social degree.

Figure 4.12: Scaling order of minimum utility ratio versus mean degrees.

This is because $\beta = -0.132$ is near zero so that the scaling order becomes close to that of the $(\alpha, 0, \bar{c})$-system, which can be shown to be equal to (4.25).

Next, we verify the scaling order of $(\alpha, 1, c)$ and $(\alpha, 0, c)$-systems versus the fraction of initial residues, $\bar{\rho}_0$, by re-visiting Fig. 4.11b. By focusing on the supercritical regime ($\bar{\rho}_0 \geq \rho_{cr}$), we observe that ρ_{min} quickly grows (with a CNE order^2) to a linear factor of $\bar{\rho}_0$. The result validates Corollary 7, in which ρ_{min} is shown to scale with respect to $\bar{\rho}_0$ multiplied by a sub-CNE^2 factor

Figure 4.13: Minimum utility ratio versus social-to-communication ratio.

$h(\bar{\rho}_0) = (1 - b_2 \exp\{-b_1[(\log(\bar{\rho}_0))^2 \langle k \rangle_c^{-1} + 2\log(\bar{\rho}_0)]\})$.[6] On the other hand, due to the same aforementioned reason (i.e., $\beta = -0.132$ is near zero), a similar scaling order can be observed for the $(\alpha, \beta, \bar{c})$-system.

Finally, we validate the scaling order in Corollary 8 by plotting ρ_{min} versus the social-to-communication ratio α in Fig. 4.13. We observe that (i) ρ_{min} increases with a CNE factor[2] of α in $(\alpha, 1, c)$ and $(\alpha, 0, c)$-systems (refer to the dotted black and solid blue plots, respectively), while (ii) it decreases linearly versus α in $(\alpha, \beta, \bar{c})$-system (see the solid red line). The latter holds because less and less number of nodes (relative to the total, N_c) depend on the social users and are disconnected initially due to S→C dependence as α increases, thus diminishing the early and subsequent stages of the network contraction process. On the other hand, the number of social users that depend on the communication nodes in (α, β, c)-systems decreases with respect to α. As a result, the portion of residual and active users at the even time steps—after taking C→S into account—also increases, thus improving the minimum utility ratio. These observations corroborate the analysis in Proposition 2.

Remark 12. *Fig. 4.13 suggests a means for identifying the root cause of network contraction, when α is increased (i.e., there are new social users joining the network). If the minimum utility ratio becomes higher, then the initial dropouts are social-based. Otherwise, they are communication-based.*

4.6 Improving the Minimum Utility Ratio

We have analyzed the minimum utility ratio ρ_{min} for networks with infinite number of nodes that, however, may not hold for practical D2D-based MSNs with *finite* number of nodes. For example, there is a gap between the theoretical and numerical results when the number of nodes, N_c, is small (see the subcritical region of Fig. 4.11). Motivated by the gap, which varies as N_c increases, we analyze how ρ_{min} will be affected by the finite N_c and then leverage the analysis to propose an admission control scheme for improving ρ_{min}.

4.6.1 Scaling Order versus Number of Nodes

To understand the minimum utility ratio in *finite networks*, i.e., D2D-based MSNs with finite number of nodes, we analyze the scaling order of ρ_{min} with respect to the number of communication nodes in Theorem 7 below. Notice that analyzing the scaling order of ρ_{min} with respect to the number of communication nodes N_c boils down to finding the expectation of the portion

[6]Note that $h(\bar{\rho}_0)$ is sub-CNE because the factors $(\log(\bar{\rho}_0))^2$ and $\log(\bar{\rho}_0)$ grow slower than $\bar{\rho}_0$.

of active node, weighted by the likelihood that a connected component exists after a network contraction process, $E[\mathbf{1}_{cc}(\mathcal{I}_0)]$. The authors in [38, SI Sec. G] simply states that the derivative of this likelihood is a Gaussian PDF with a standard deviation proportional to $\frac{1}{\sqrt{N_c}}$. Instead of using this result directly, we provide a more elaborate derivation using Lindeberg's theorem [33] to show the exact PDF of $E[\mathbf{1}_{cc}(\mathcal{I}_0)]$. Consequently, we obtain a clear scaling order versus N_c, which is stated as follows.

Theorem 7. *The scaling order of minimum utility ratio versus the number of communication nodes N_c satisfies*

$$\rho_{min} = \begin{cases} \Theta\left(-\sqrt{1-\exp\{-b_4 N_c\}}\right), & \text{for } \bar{\rho}_0 < \rho_{cr}, \\ \Theta\left(-\exp\{-b_5 N_c\}\right), & \text{for } \bar{\rho}_0 \geq \rho_{cr}, \end{cases} \tag{4.27}$$

where $b_4 = \sigma^2(\bar{\rho}_0 - \rho_{cr})^2$ and $b_5 = \frac{2}{3\sigma^2}(\bar{\rho}_0 - \rho_{cr})^2$ for (α, β, c)-system, with $\sigma^2 := Var[\mathbf{1}_{(v \notin \mathcal{I}_0)}]$ denotes the variance of a $(0/1)$-process that indicates whether a node v is dropped out initially. On the other hand, these constants are equal to $b_4 = \alpha\sigma^2(\bar{\rho}_0 - \rho_{cr})^2$ and $b_5 = \frac{2\alpha}{3\sigma^2}(\bar{\rho}_0 - \rho_{cr})^2$ for $(\alpha, \beta, \bar{c})$-system

Proof. **(α, β, c)-system in the subcritical regime.** For this regime, we are interested in how a finite N_c will induce a non-zero ρ_{min}. Consider one realization of initial node dropouts, $\mathcal{I}_0 \subset \mathcal{V}$, and denote $\rho_a := \frac{|\mathcal{V} \setminus \mathcal{I}_0|}{N_c}$ as the *actual* fraction of initial dropouts. For this particular $\mathcal{I}_0$, ρ_{min} can be re-defined as the smallest utility ratio averaged over many D2D-based MSN graph snapshots, which is equal to the theoretical result in Fig. 4.11 after re-labeling the horizontal axis to ρ_a. In this figure, one or more active nodes will always exist and ρ_{min} is non-zero if $\rho_a > \rho_{cr}$. Thus, we are interested in finding the likelihood $\Pr\{\rho_a > \rho_{cr}\}$.

We start by analyzing the probability density function (PDF) of ρ_a as follows. Let $\mathbf{1}(v) := \mathbf{1}_{(v \notin \mathcal{I}_0)}$ be a $(0/1)$-function indicating node $v \in \mathcal{V}$ is *not* dropped out initially, which has an expected value and a variance of $\bar{\rho}_0$ and $\sigma^2 := Var[\mathbf{1}(v)]$, respectively. Denote $\mathcal{S} := \{\mathbf{1}(v_1), \ldots, \mathbf{1}(v_{N_c})\}$ as a sequence that has a sample mean of $\rho_a := \frac{1}{N_c}\sum_{v \in \mathcal{V}} \mathbf{1}(v)$. In general, the elements of $\mathcal{S}$ are not identically-distributed, especially under degree-weighted initial dropouts (when $\beta = 1$) where $\mathbf{1}(v)$ depends on the communication degree of node v. However, since the elements are bounded above by one, Lindeberg's triangular array theorem [33, ex. 27.4] states that the sample mean converges to $\rho_a \sim \mathcal{N}(\bar{\rho}_0, \frac{\sigma^2}{N_c})$, which is a Gaussian distribution with mean $\bar{\rho}_0$ and variance proportional to $\frac{1}{N_c}$.

We proceed by calculating the likelihood

$$\Pr\{\rho_a > \rho_{cr}\} = \int_{x=\rho_{cr}}^{\infty} p(x)d(x) = \frac{1}{2}\left[1 + \text{erf}\left(\frac{\bar{\rho}_0 - \rho_{cr}}{\sigma\sqrt{2/N_c}}\right)\right], \tag{4.28}$$

where $p(x)$ is the PDF of ρ_a and $\text{erf}(\cdot)$ is a Gaussian error function. Denote $z := \frac{(\bar{\rho}_0 - \rho_{cr})\sqrt{N_c}}{\sigma\sqrt{2}}$, which is smaller than 0 because $\bar{\rho}_0 < \rho_{cr}$, and apply the approximation [17] $\text{erf}(z) \approx \text{sgn}(z)(1 - \exp\{-z^2\frac{4/\pi + az^2}{1+az^2}\})^{1/2}$, where $a := \frac{8(\pi-3)}{3\pi(4-\pi)}$ and $\text{sgn}(z)$ is equal to -1 when z is negative and +1, otherwise. By noticing that $\frac{4/\pi+az^2}{1+az^2} \approx 1$, the likelihood in (4.28) can be re-stated as

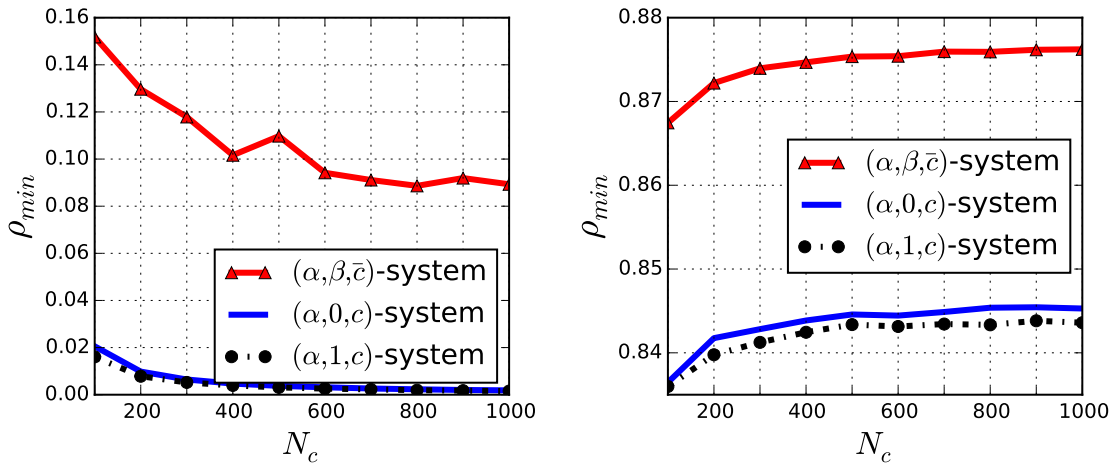
$$\Pr\{\rho_a > \rho_{cr}\} \approx \frac{1}{2}\left[1 - \sqrt{1 - \exp\left\{-\frac{(\rho_{cr} - \bar{\rho}_0)^2 N_c}{2\sigma^2}\right\}}\right]. \tag{4.29}$$

With (4.29) at hand, we are now ready to analyze the scaling order of ρ_{min} with respect to N_c. Let $\mathbf{1}_{cc}(\mathcal{I}_0)$ indicates whether a connected component exists at the end of a network contraction process triggered by $\mathcal{I}_0$. In other words, $\mathbf{1}_{cc}(\mathcal{I}_0) = 1$ if ρ_{min} is non-zero. Denote $\rho_{min}(\mathcal{I}_0)$ as the minimum utility ratio corresponding to $\mathcal{I}_0$, while $E[\rho_{min}(\mathcal{I}_0)|\mathbf{1}_{cc}(\mathcal{I}_0)]$ is the average over all possible $\mathcal{I}_0$'s corresponding to a mean fraction of initial residues $\bar{\rho}_0$, given that ρ_{min} is non-zero. Then, the minimum utility ratio as a function of N_c is given as

$$\begin{aligned}
\rho_{min}(N_c) &= E[\rho_{min}(\mathcal{I}_0)|\mathbf{1}_{cc}(\mathcal{I}_0)] \times E[\mathbf{1}_{cc}(\mathcal{I}_0)] &(4.30)\\
&= E[\rho_{min}(\mathcal{I}_0)|\mathbf{1}_{cc}(\mathcal{I}_0)] &(4.31)\\
&\quad \times \frac{1}{2}\left[1 - \sqrt{1 - \exp\left\{-\frac{(\rho_{cr} - \bar{\rho}_0)^2}{2\sigma^2} N_c\right\}}\right],
\end{aligned}$$

where $E[\mathbf{1}_{cc}(\mathcal{I}_0)]$ is equal to (4.29) by definition and $E[\rho_{min}(\mathcal{I}_0)|\mathbf{1}_{cc}(\mathcal{I}_0)]$ is a constant with respect to N_c. After applying Definition 19; dropping the constant terms before the square-root term in (4.31); and setting $b_4 := \frac{1}{2\sigma^2}(\bar{\rho}_0 - \rho_{cr})^2$, we finally obtain the first line of (4.27).

(α, β, c)-system in the supercritical regime. The steps of proof are mainly similar to that for the subcritical regime. The main difference is that now $z = \frac{(\bar{\rho}_0 - \rho_{cr})\sqrt{N_c}}{\sigma\sqrt{2}}$ is *larger*. To proceed, we apply the Gaussian error function approximation $\text{erf}(z) \approx 1 - (\frac{1}{6}\exp(-z^2) +$



(a) Subcritical region ($\bar{\rho}_0 < \rho_{cr}$). (b) Supercritical region ($\bar{\rho}_0 \geq \rho_{cr}$).

Figure 4.14: Minimum utility ratio versus number of communication nodes.

$\frac{1}{2}\exp(\frac{4}{3}z^2)$) [17]. Then, the minimum utility ratio becomes

$$\begin{aligned}
&\rho_{min}(N_c) \\
&= E[\rho_{min}(\mathcal{I}_0)|\mathbf{1}_{cc}(\mathcal{I}_0)] \times E[\mathbf{1}_{cc}(\mathcal{I}_0)] \qquad (4.32)\\
&= E[\rho_{min}(\mathcal{I}_0)|\mathbf{1}_{cc}(\mathcal{I}_0)] \times \left[1 - \left(\frac{1}{12}\exp\left\{-\frac{(\rho_{cr}-\bar{\rho}_0)^2 N_c}{2\sigma^2}\right\} + \frac{1}{4}\exp\left\{-\frac{4(\rho_{cr}-\bar{\rho}_0)^2 N_c}{6\sigma^2}\right\}\right)\right],
\end{aligned}$$

where $E[\rho_{min}(\mathcal{I}_0)|\mathbf{1}_{cc}(\mathcal{I}_0)]$ corresponds to ρ_{min} as the solution of (4.13), which is constant with respect to N_c. After applying Definition 19, dropping the constant terms, and noticing that the first exponential term grows slower than the second one, we get $\rho_{min} = \Theta(\frac{1}{2}\exp\left\{-\frac{4(\rho_{cr}-\bar{\rho}_0)^2}{6\sigma^2}N_c\right\})$, which can easily be reduced to the second line of (4.27).

$(\alpha, \beta, \bar{c})$-system in the sub- and super-critical regimes. The scaling order for $(\alpha, \beta, \bar{c})$-system can be analyzed by following the exact same steps as in the proofs above, but with applying a social-based initial dropouts ($\mathcal{I}_0 \subset \mathcal{V}_s$) and replacing N_c with the number of social nodes $N_s = \alpha N_c$. □

4.6.2 Numerical Validation

Next, we validate the scaling order in Theorem 7 via numerical simulations. In our simulations, we vary the number of communication nodes N_c from 100 to 1,000, set $\alpha = 0.8$, and assign the number of social nodes to $N_s = \alpha N_c$. To maintain consistency with Section 4.5.4, we re-employ the ER graph model with the average degrees of $\langle k\rangle_c = 6$ and $\langle k\rangle_s = 10$.

We consider the subcritical case ($\bar{\rho}_0 < \rho_{cr} = 0.411$) by setting the mean fraction of initial

dropouts to $\bar{\rho}_0 = 0.15$. According to Theorem 7, the minimum utility ratio should *decrease* according to $\rho_{min} \sim -\sqrt{1-\exp\{-b_4 N_c\}}$, which can be reduced to $\rho_{min} \sim \exp\{-b_4 N_c\}$ since the curve shape of $1-\exp\{-b_4 N_c\}$ will not be changed by a square-root operator [156]. This reduced scaling order fits the numerical result in Fig. 4.14a well, indicating the first line of (4.27) is valid. We also consider the supercritical regime by setting the mean fraction of initial dropouts to $\bar{\rho}_0 = 0.85$. According to the resulting plots for this regime in Fig. 4.14b, ρ_{min} *increases* with a negative exponential factor of N_c, matching the second line of (4.27) and completing our validation.

4.6.3 Maximizing the Minimum Utility Ratio

After validating our analysis, we discuss how the scaling order of minimum utility ratio ρ_{min} in Theorem 7 can be used to maintain the performance of D2D-based MSNs. It is desirable to have a large ρ_{min} so that many users can still utilize the network for exchanging social contents. Let there be new users that want to join the D2D-based MSN for benefits such as social networking with the existing users and/or saving cellular data access fee [4]. In this case, we can apply *admission control* [94], which is a validation process applied before a new user can join the network. Such a control can be enforced by allowing/preventing new downloads of app for accessing the D2D-based MSN, or by dedicating one or more nodes for validating new users. The main question is, while trying to maximize ρ_{min}, how to decide whether new users should be allowed to join the D2D-based MSN? To answer this question, we propose a minimum utility ratio-based (`MinUtil`) admission scheme, which is detailed as follows.

Minimum Utility Ratio-based Admission Control

The proposed `MinUtil` scheme first estimates the critical fraction of initial residues ρ_{cr} by solving (4.13), and then determines the mean fraction of initial residues $\bar{\rho}_0$ (e.g., by taking a sample mean of the number of initial dropouts reported by a subset of nodes called detectors [94]). Then, `MinUtil` leverages the analytical result in Theorem 7 to maximize the minimum utility ratio. Specifically,

1. the proposed scheme will not admit new users under severe initial dropouts (i.e., when $\bar{\rho}_0 < \rho_{cr}$);
2. otherwise, `MinUtil` allows all new users to join the D2D-based MSN when $\bar{\rho}_0 \geq \rho_{cr}$.

Performance Evaluation

To evaluate the proposed `MinUtil` scheme, we generate 10^3 realizations of a D2D-based MSN graph $\mathcal{G} = (\mathcal{E}_c, \mathcal{E}_s)$—called the *base graph*—with $N_c = 100$ communication nodes and the

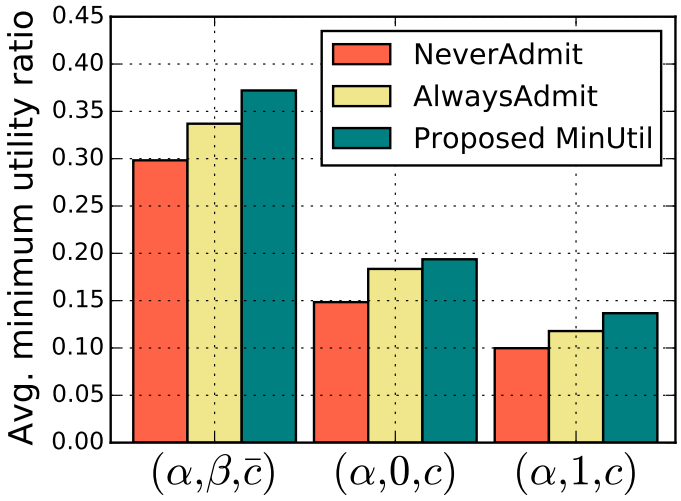


Figure 4.15: Performance evaluation of the proposed `MinUtil` scheme.

average degrees of $\langle k \rangle_c = 6$ and $\langle k \rangle_s = 10$. We consider the case where a set of new nodes $\mathcal{V}_{new}$ ($\mathcal{V} \cap \mathcal{V}_{new} = \emptyset$), which all act as both communication nodes and social users, want to join the D2D-based MSN. Upon being admitted, each node in $\mathcal{V}_{new}$ forms a communication and a social link to each existing node with probabilities $\frac{\langle k \rangle_c}{N_c}$ and $\frac{\langle k \rangle_s}{N_s}$, respectively, to ensure that the resulting *evolved graph* $\mathcal{G}' = (\mathcal{V} \cup \mathcal{V}_{new}, \mathcal{E}'_c, \mathcal{E}'_s)$ still consists of ER communication and social graphs. We set the fraction of newly-joining nodes to $\frac{\mathcal{V}_{new}}{N_c} = 10\%$, while the initial dropout ratio is selected uniformly at random from the range $\bar{\rho}_0 \in [0.01, 0.5]$ to emulate mild to moderate dropout severity. We generate 10 different combinations of $(\mathcal{V}_{new}, \bar{\rho}_0)$ for each realization of $\mathcal{G}$ (correspondingly, $\mathcal{G}'$), and further select 100 different sets of initial dropouts, $\{\mathcal{I}_0\}$, for each $(\mathcal{V}_{new}, \bar{\rho}_0)$.

We compare the proposed `MinUtil` admission to the schemes where new users $\mathcal{V}_{new}$ are always admitted (`AlwaysAdmit`) and never admitted (`NeverAdmit`). In `AlwaysAdmit` (resp., `NeverAdmit`), the minimum utility ratio, ρ_{min}, is computed by applying a network contraction process to the evolved graph $\mathcal{G}'$ (resp., base graph $\mathcal{G}$). On the other hand, the `MinUtil` scheme can either use $\mathcal{G}$ or $\mathcal{G}'$, depending on whether new users are admitted according to criteria 1) and 2) above.

The *average minimum utility ratio* $E(\rho_{min})$ of the aforementioned schemes are depicted in Fig. 4.15. Recall that the `NeverAdmit` scheme tries to keep N_c low and benefits from the sub-critical regime ($\bar{\rho}_o < \rho_{cr}$), which unfortunately contributes to only a small portion of $E(\rho_{min})$. Recall that according to our results in Section 4.5.4 the critical initial dropout ratio ρ_{cr} is equal to 0.2, 0.3, and 0.411 for (α, β, c), $(\alpha, 0, c)$, and $(\alpha, 1, c)$-systems, respectively. In contrast, the `AlwaysAdmit` scheme is more beneficial when $\bar{\rho}_0 \geq \rho_{cr}$, which generates a higher ρ_{min} (see

Fig. 4.14b) and contributes more towards $E(\rho_{min})$, thus producing a higher (better) $E(\rho_{min})$ in Fig. 4.15. *The proposed* `MinUtil` *scheme, on the other hand, exhibits the largest the largest average minimum utility ratio* since it can leverage the opportunity to increase ρ_{min} by minimizing (resp., maximizing) the number of nodes in the subcritical (resp., supercritical) regime.

With respect to the cause of initial dropouts, $(\alpha, \beta, \bar{c})$-system has the largest $E(\rho_{min})$ among other systems in Fig. 4.15, matching the results in Figs 4.10-4.14. This is because the supercritical case that produces high ρ_{min} is more likely to occur in the $(\alpha, \beta, \bar{c})$-system, that is, about 60.2% of the time. Finally, we also observe that the $(\alpha, 1, c)$-system offers the largest relative $E(\rho_{min})$ improvement of up to 36.74% when the `MinUtil` scheme is employed.

4.7 Node Resilience in D2D-based MSN

We have discussed that D2D-based MSN users may become non-active due to either (i) the initial dropouts at time $t = 0$, or (ii) a dropout-induced network contraction process at $t \geq 1$. On the other hand, these users may require a non-negative *session time*, which denotes the time until they respectively finish a social content exchange through the D2D-based MSN. It is desirable that each user can enjoy a stable network connection during his/her session time, in that the D2D-based MSN is said to be able to provide a *resilient* networking service.

The network-wide resilience of wireless systems against node failure [56, 29] and misbehavior [162, 163] has been widely studied in the literature. However, little is known about how wireless network can provide resilient service from end-users' perspective, a quality-of-experience metric that is vital to the success of D2D-based MSN's deployment. Therefore, we are interested in analyzing the D2D-based MSN's resilience, viewed from users' perspective.

4.7.1 Definition of Node Resilience

Before studying the resilience from users' perspective, we define a deactivation time metric as follows.

Definition 20. *Deactivation time of user $v \in \mathcal{V}_s$ is the smallest even time step in which v becomes non-active in the D2D-based MSN, i.e.,*

$$\tau_v := \inf\{t : v \in \mathcal{F}_s(t-1), v \notin \mathcal{F}_s(t)\}. \tag{4.33}$$

We choose even (two) time steps as the time granularity to make sure that τ_v captures the impacts of both the C→S and S→C dependencies, which occur alternatively in two consecutive steps. Moreover, we say that a node is *affected* if it becomes isolated after a cascade-of-failures.

An immediate question is how to relate the discrete time steps of τ_{max} to the real time (in milliseconds). To answer this, we first consider the S→C dependence at step $t = 3$ in Fig. 4.4,

where node v_4 is previously dropped out. In this case, nodes v_5 and v_6 will wait for a periodic status update from v_4 as their communication neighbor. When the neighbor does not respond within a preset time interval T_c, then v_5 and v_6 conclude that v_4 has been dropped out,[7] and v_5 immediately becomes non-active due to isolation.

Next, we consider a C→S dependence at step $t = 4$ in Fig. 4.4. In this case, v_6 is already aware that the physical path to reach its social neighbor v_5 is unavailable and immediately becomes non-active in the social graph. This indicates the C→S dependence occurs instantaneously. As a result, *two* consecutive time steps in τ_v is equal to T_c due to the S→C dependence only. For simplicity, we assume that *one* discrete step is effectively equal to $\frac{T_c}{2}$ milliseconds.

To proceed, let a *session* denote an exchange of social contents between between two D2D-based MSN users, and a *session time* $ST_v \in (0, \infty)$ be the amount of time needed to complete a session initiated by node $v \in \mathcal{V}$. This metric is normalized to the length of one time step, $\frac{T_c}{2}$, and can either be heavy-tailed or exponentially-distributed [163]. We consider the *worst-case* session time requirement by assuming all the sessions under consideration starts at time $t = 0$, that is, right when the initial dropouts occur. Moreover, we assume that a node will not start a new session if it has detected any failure in the network.

With the aforementioned assumptions and definitions, we are ready to define a node resilience metric as follows.

Definition 21. *Node resilience is the likelihood that a randomly-chosen node $v \in \mathcal{V}$ does not become non-active during its ongoing session, denoted as*

$$\Psi_n := \Pr\{ST_v < \tau_v\}. \tag{4.34}$$

Node resilience metric Ψ_n quantifies the D2D-based MSN's ability to support an ongoing session and provides a satisfaction measure from end-users' perspective.

4.7.2 Bounds of Node Resilience Metric

The node resilience Ψ_n in (4.34) depends on the deactivation time τ_v that, unfortunately, is hard to derive in closed-form. Thus, instead of examining τ_v directly, we focus on the affected nodes and the upper bound of their deactivation times as follows.

Definition 22. *Maximum deactivation time is the expected largest number of time steps until an affected node becomes deactivated, that is,*

$$\tau_{max} := E[\sup\{\tau_v : v \notin \lim_{t\to\infty} \mathcal{F}_s(t)\}]. \tag{4.35}$$

[7] For example, T_c is equal to (`hello_interval*allowed_hello _loss`) milliseconds when AODV routing protocol [118] is employed.

In other words, the maximum deactivation time τ_{max} quantifies the smallest number of time steps until no more node (equivalently, less than one node) becomes non-active, and can be re-stated as

$$\tau_{max} = \inf\left\{ t \geq 1 \,\middle|\, \bar{\rho}(t) - \bar{\rho}(t-2) < \frac{1}{N} \right\}, \tag{4.36}$$

with N equals N_s and N_c for (α, β, c) and $(\alpha, \beta, \bar{c})$-systems, respectively.

According to (4.36), the maximum deactivation time for (α, β, c)-system can be determined by setting $\bar{\chi}_c(1) = \bar{\rho}_0$, calculating $\bar{\rho}(t) = \bar{\chi}_s(t)\psi_s(\bar{\chi}_s(t))$, and finding the smallest t such that $\bar{\rho}(t) - \bar{\rho}(t-2) < \frac{1}{N_s}$. For $(\alpha, \beta, \bar{c})$-system, we can set $\bar{\chi}_s(1) = \bar{\rho}_0$, compute $\bar{\rho}(t)$ using Lemma 10, and find $\tau_{max} = \arg\max\{\bar{\rho}(t) - \bar{\rho}(t-2) < \frac{1}{N_c}\}$. The corresponding $\bar{\rho}(t)$ when τ_{max} is achieved is equal to the minimum utility ratio, that is, $\rho_{min} = \bar{\rho}(\tau_{max})$.

With τ_{max} and ρ_{min} at hand, we can calculate the lower and upper bounds of Ψ_n using the following theorem.

Theorem 8. *For Pareto-distributed ST_v with PDF $Pr\{ST_v = x\} = \frac{\gamma x_m^\gamma}{x^{\gamma+1}}$, shape parameter $\gamma > 0$, and scale $x_m > 0$, the node resilience is lower and upper-bounded by*

$$\rho_{min} + (\bar{\rho}_0 - \rho_{min})(1 - x_m^\gamma) \leq \Psi_n \leq (1 - \bar{\rho}_0 + \rho_{min})(x_m/\tau_{max})^\gamma - (1 - \bar{\rho}_0). \tag{4.37}$$

Proof. Finding the lower and upper bounds of Ψ_n is equivalent to the problem of analyzing the CDF of session time ST_v outlined as follows. Let $\mathcal{C}_{net} := \mathcal{V} \setminus (\lim_{t\to\infty} \mathcal{F}_s(t) \cup \mathcal{I}_0)$ be the set of users that become non-active due to the network contraction process at time $t \geq 1$. Then, we have

$$\begin{aligned} Pr\{ST_v > \tau_v\} &= Pr\{ST_v > \tau_v | v \in \mathcal{I}_0\} Pr\{v \in \mathcal{I}_0\} \\ &\quad + Pr\{ST_v > \tau_v | v \in \mathcal{C}_{net}\} Pr\{v \in \mathcal{C}_{net}\} \\ &\quad + Pr\{ST_v > \tau_v | v \notin \mathcal{C}_{net} \cup \mathcal{I}_0\} Pr\{v \notin \mathcal{C}_{net} \cup \mathcal{I}_0\}. \end{aligned} \tag{4.38}$$

When node u is not affected by the initial dropouts nor the network contraction process (i.e., $v \notin \mathcal{I}_0 \cup \mathcal{C}_{net}$), then its session time ST_v is always lower than the isolation time τ_v, which is equal to ∞ by definition. Thus, the last summation term becomes zero and (4.38) can be re-stated as

$$\begin{aligned} &Pr\{ST_v > \tau_v | v \in \mathcal{I}_0\} Pr\{v \in \mathcal{I}_0\} \\ &\quad + Pr\{ST_v > \tau_v | v \in \mathcal{C}_{net}\} Pr\{v \in \mathcal{C}_{net}\} \qquad (4.39) \\ &\leq (1 - \bar{\rho}_0) + (\bar{\rho}_0 - \rho_{min}) Pr\{ST_v > \tau_{max}\}, \qquad (4.40) \end{aligned}$$

where (4.40) holds because $\tau_v = 0$ when node u belongs to the set of initial dropouts (i.e.,

$v \in \mathcal{I}_0$) such that $Pr\{ST_v > \tau_v | v \in \mathcal{I}_0\} = 1$, and $Pr\{ST_v > \tau_v | v \in \mathcal{C}_{net}\} \leq Pr\{ST_v > \tau_{max}\}$ because the node deactivation due to network contractions always happen before the maximum deactivation time τ_{max}. By the fact that $\Psi_n = 1 - Pr(ST_v > \tau_v)$ and after plugging in the CCDF of the Pareto-distributed ST_v, we obtain the upper bound in (4.43).

For the lower bound, we re-state the node resilience as

$$\begin{aligned}\Psi_n &= Pr\{ST_v < \tau_v\} \\ &= Pr\{ST_v < \tau_v | v \in \mathcal{I}_0\} Pr\{v \in \mathcal{I}_0\} + Pr\{ST_v < \tau_v | v \in \mathcal{C}_{net}\} Pr\{v \in \mathcal{C}_{net}\} \\ &\quad + Pr\{ST_v < \tau_v | v \notin \mathcal{C}_{net} \cup \mathcal{I}_0\} Pr\{v \notin \mathcal{C}_{net} \cup \mathcal{I}_0\}.\end{aligned} \tag{4.41}$$

When $v \in \mathcal{I}_0$, the isolation time is always equal to $\tau_{max} = 0$ by definition, such that the first summation term disappears and the node resilience Ψ_n can be re-stated as

$$\begin{aligned}&Pr\{ST_v < \tau_v | v \in \mathcal{C}_{net}\} Pr\{v \in \mathcal{C}_{net}\} + Pr\{ST_v < \tau_v | v \notin \mathcal{C}_{net} \cup \mathcal{I}_0\} Pr\{v \notin \mathcal{C}_{net} \cup \mathcal{I}_0\} \\ &\quad \geq \rho_{min} + (\bar{\rho}_0 - \rho_{min}) Pr\{ST_v < 1\},\end{aligned} \tag{4.42}$$

where (4.42) applies because $Pr\{ST_v < \tau_v | v \in \mathcal{C}_{net}\} \leq Pr\{ST_v < 1\}$, that is, the node dropouts due to network contractions always happen at $t \geq 1$. □

By following the steps in the proof of Theorem 8 and employing the CCDF of the exponential distribution [43], we obtain the following corollary.

Corollary 9. *For exponentially-distributed node session times with PDF $Pr\{ST_v = x\} = \lambda e^{-\lambda x}$ and mean $\frac{1}{\lambda} > 0$, the node resilience is bounded by*

$$\rho_{min} + (\bar{\rho}_0 - \rho_{min})(1 - e^{-\lambda}) \leq \Psi_n \leq (1 - \bar{\rho}_0) + (\bar{\rho}_0 - \rho_{min}) e^{-\lambda \tau_{max}}.$$

Remark 13. *The results in Theorem 8 and Corollary 9 can easily be extended to the more general cases by substituting $1 - Pr\{ST_v < 1\}$ and $Pr\{ST_v > \tau_{max}\}$ with the CCDFs of the session time ST_v under examination.*

4.7.3 Scaling Order of Node Resilience Metric

Next, we will analyze the impacts of networks parameters (i.e., $\bar{\rho}_0$ and N_c) to the node resilience Ψ_n, which is hard to do using the bounds in (4.43) and (4.37) that depend on the numerical calculations of τ_{max} and ρ_{min}. Instead, we study the impacts of $\bar{\rho}_0$ and N_c through the scaling order of Ψ_n.

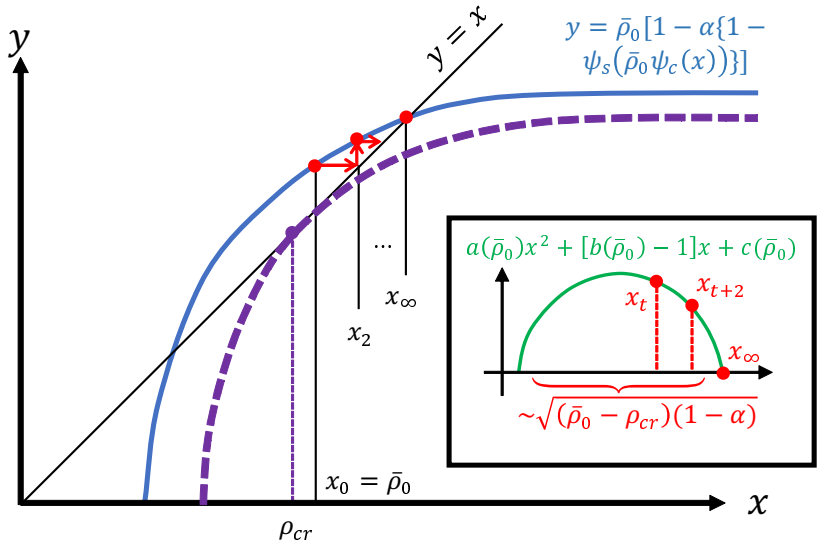


Figure 4.16: Illustration for the proof of Lemma 12 when $\bar{\rho}_0 \geq \rho_{cr}$.

Since according to (4.43) and (4.37) Ψ_n is a function of τ_{max}, then we present the following scaling order result for τ_{max}.

Lemma 12. *For the supercritical region ($\bar{\rho}_0 \geq \rho_{cr}$), the scaling order of maximum deactivation time satisfies*

$$\tau_{max} = \begin{cases} \Theta\left(\frac{\log(N_c)}{\sqrt{(1-\alpha)(\bar{\rho}_0-\rho_{cr})}}\right), & \textit{for } (\alpha,\beta,c)\textit{-system}, \\ \Theta\left(\frac{\log(\alpha N_c)}{\sqrt{(1-\alpha)(\bar{\rho}_0-\rho_{cr})}}\right), & \textit{for } (\alpha,\beta,\bar{c})\textit{-system}, \end{cases} \tag{4.43}$$

while for the subcritical region ($\bar{\rho}_o < \rho_{cr}$),

$$\tau_{max} = \Theta\left([(1-\alpha)(\rho_{cr}-\bar{\rho}_0)]^{-1/2}\right). \tag{4.44}$$

Proof. **(α, β, c)-system in the supercritical regime.** The maximum deactivation time τ_{max} is equivalent to the instance when $\bar{\rho}(t)$ converges to ρ_{min}. According to (4.4), such a convergence occurs when the following equation holds.

$$\bar{\chi}_c(t) = \bar{\rho}_0\left(1 - \alpha\left[1 - \psi_s\left(\bar{\rho}_0 \times \psi_c(\bar{\chi}_c(t-1))\right)\right]\right). \tag{4.45}$$

Thus, we plot the right side of (4.45) with $\bar{\chi}_c(t-2) = x$ as the blue curve in Fig. 4.16 and the left side with $\bar{\chi}_c(t) = x$ as the black diagonal plot. The former intersects with the latter plot

at two non-trivial points, since we are considering the supercritical regime ($\bar{\rho}_0 \geq \rho_{cr}$). To find τ_{max}, we start by setting $\bar{\chi}_c(0) = \bar{\rho}_0$ (see point x_0 in Fig. 4.16). Let an *iteration* be a one-step calculation of $\bar{\chi}_c(t+2)$ from $\bar{\chi}_c(t)$ by using (4.45), which is represented as a horizontal red arrow from the blue to the black curve in Fig. 4.16). Then, τ_{max}, which is the number of iterations until $\bar{\chi}_c(t-2)$ and $\bar{\chi}_c(t)$ in (4.45) converges to each other, is proportional to the number of arrows from x_0 until the blue and black plots meets at x_∞.

To proceed, we assume that $\bar{\rho}_0$ is near ρ_{cr} such that the right side of (4.45), $f(x) := \bar{\rho}_0(1-\alpha[1-\psi_s\left(\bar{\rho}_0\psi_c(\bar{\chi}_c(t-1))\right)])|_{\bar{\chi}_c(t-2)=x}$, can be approximated into the quadratic form

$$f(x) \approx a(\bar{\rho}_0)x^2 + b(\bar{\rho}_0)x + c(\bar{\rho}_0). \tag{4.46}$$

Let $f_{cr}(x) := f(x)|_{\bar{\rho}_0=\rho_{cr}}$ be the right side of (4.45) that corresponds to the critical point when ρ_{min} starts to be non-zero, and is plotted as a purple curve in Fig. 4.16. Now, the vertical distance between the quadratic function (4.46) and the left side of (4.45) is approximately equal to the vertical distance between (4.46) and $f_{cr}(x)$, which amounts to $(1-\alpha)(\bar{\rho}_0 - \rho_{cr})$. This implies the horizontal distance between the starting point x_0 and the destination (convergence) point x_∞ is proportional to $\sqrt{(1-\alpha)(\bar{\rho}_0-\rho_{cr})}$, as indicated by the inset in Fig. 4.16. Moreover, since $f(x)$ is convex, its derivative is proportional to $-\sqrt{(1-\alpha)(\bar{\rho}_0-\rho_{cr})}$. According to Lemma 11, ψ_c and ψ_s are functions of the degree distributions, which have exponential forms (see Figs. 4.7 and 4.6). Consequently, the iterations converge to the solution of x_∞ as $f' = \exp\left\{-\tau_{max}\sqrt{(1-\alpha)(\bar{\rho}_0-\rho_{cr})}\right\}$, where τ_{max} is the number of stair treads (steps). These iterations will stop when there is less than one dropped out nodes, that is, when $\bar{\chi}_c(t-2) - \bar{\chi}_c(t) < \frac{1}{N_c}$. As a result, the stopping condition can be stated as $f' \approx \frac{1}{N_c}$ and we obtain $\tau_{max} \sim \frac{\log(N_c)}{\sqrt{(1-\alpha)(\bar{\rho}_0-\rho_{cr})}}$.

$(\alpha, \beta, \bar{c})$-system in the supercritical regime. Finding τ_{max} is equivalent to counting the number of iterations until (4.10) converges. Notice that (4.10) can be re-stated as (4.45), but with $\bar{\chi}_c(t)$, $\bar{\chi}_c(t-2)$, and $\bar{\chi}_c(0)$ replaced by $\bar{\chi}_s(t)$, $\bar{\chi}_s(t-2)$, and $\bar{\chi}_s(0) = \bar{\rho}_0$, respectively. Thus, we can re-apply the same steps as above and find that the iterations converge to the solution $x_\infty = \bar{\chi}_s(t)$ as $f' = \exp\left\{-\tau_{max}\sqrt{(1-\alpha)(\bar{\rho}_0-\rho_{cr})}\right\}$, with a stopping condition of $f' \sim \frac{1}{N_c} = \frac{1}{\alpha N_c}$. We finally obtain $\tau_{max} \sim \frac{\log(\alpha N_c)}{\sqrt{(1-\alpha)(\bar{\rho}_0-\rho_{cr})}}$, which according to Definition 19 can be re-stated as the second line of (4.43).

(α, β, c)-system in the subcritical regime. In this regime, the left and right sides of (4.4) intersect each other at a trivial point $x = 0$. Similar to Fig. 4.16, one iteration of calculating $\bar{\chi}_c(t)$ from $\bar{\chi}_c(t-2)$ using (4.45) can be represented as a horizontal red arrow in Fig. 4.17, but now the arrow points to the left and the convergence point x_∞ is at zero. Our objective is now to find τ_{max} by counting the number of iterations (arrows) from $x_0 = \bar{\chi}_c(0) = \bar{\rho}_0$ to $x_\infty = \bar{\chi}_c(\infty) = 0$.

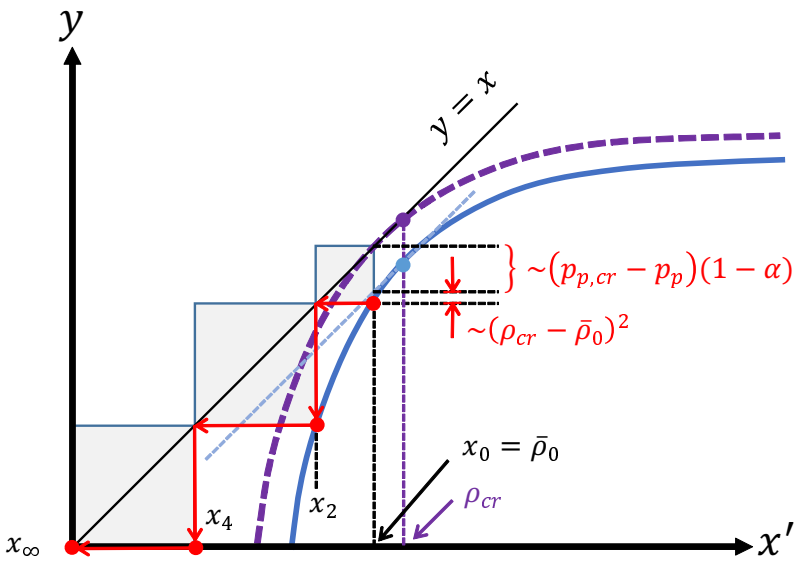


Figure 4.17: Illustration for the proof of Lemma 12 when $\bar{\rho}_0 < \rho_{cr}$.

We proceed by re-employing the quadratic approximation in (4.46) to the right side of (4.45), and examining the vertical distance between the black plot and the starting point $x_0 = \bar{\rho}_0$. According to Fig. 4.17, this is equal to a small vertical distance proportional to $(\rho_{cr} - x_0)^2$, plus the gap between $f(x)$ as the right side of (4.45) and $f_{cr}(x) := f(x)|_{\bar{\rho}_0 = \rho_{cr}}$. This gap is equal to $(1-\alpha)(\rho_{cr} - \bar{\rho}_0)$. Since the black plot representing the left side of (4.45) has a unity gradient, then its horizontal distance to x_0 is proportional to

$$(1-\alpha)(\rho_{cr} - \bar{\rho}_0)^2 + (1-\alpha)(\rho_{cr} - \bar{\rho}_0). \tag{4.47}$$

For every small horizontal distance of dx_0, the number of steps to be taken is proportional to $\frac{dx_0}{(\rho_{cr}-x_0)^2+(\rho_{cr}-\bar{\rho}_0)(1-\alpha)}$. Notice that the total horizontal distance to be traveled by all iterations (arrows) is equal to $\bar{\rho}_0$, so that the total number of required steps becomes

$$\tau_{max} \sim \int_{x_0=0}^{\bar{\rho}_0} \frac{dx_0}{(\rho_{cr} - x_0)^2 + (\rho_{cr} - \bar{\rho}_0)(1-\alpha)} \tag{4.48}$$

$$= \frac{\pi/2}{\sqrt{(1-\alpha)(\bar{\rho}_0 - \rho_{cr})}}, \tag{4.49}$$

where (4.49) can be acquired after applying [176, Sec. 2.103] to (4.48) and assuming that $\bar{\rho}_0$ is near ρ_{cr}. Finally, (4.44) is obtained by using Definition 19 and dropping constant $\pi/2$ from (4.49).

$(\alpha, \beta, \bar{c})$-system in the subcritical regime. Recall that finding τ_{max} is equivalent to

counting the number of iterations until (4.10) converges, while (4.10) can be re-stated as (4.45), but with $\bar{\chi}_c(t)$, $\bar{\chi}_c(t-2)$, and $\bar{\chi}_c(0)$ replaced by $\bar{\chi}_s(t)$, $\bar{\chi}_s(t-2)$, and $\bar{\chi}_s(0) = \bar{\rho}_0$, respectively. Thus, we can follow the steps in the proof for (α, β, c)-system under the subcritical regime above, and finally obtain the scaling order in (4.44). □

With Lemma 12 at hand, we are ready to analyze the scaling order of Ψ_n as follows.

Theorem 9. *For a D2D-based MSN with Pareto-distributed session times, the scaling order of the node resilience Ψ_n versus the initial survival probability $\bar{\rho}_0$ satisfies*

$$\Psi_n = \begin{cases} O\left(-\bar{\rho}_0(\bar{\rho}_0 - \rho_{cr})^{\gamma/2}\right), & \text{for } \bar{\rho}_0 \geq \rho_{cr}, \\ O\left(-\bar{\rho}_0(\rho_{cr} - \bar{\rho}_0)^{\gamma/2}\right), & \text{for } \bar{\rho}_0 < \rho_{cr}, \end{cases} \tag{4.50}$$

while the scaling order of Ψ_n versus the number of communication nodes N_c for the supercritical regime ($\bar{\rho}_0 \geq \rho_{cr}$) satisfies

$$\Psi_n = O\left(\log(N_c)\right). \tag{4.51}$$

Proof. To prove (4.50), we vary $\bar{\rho}_0$ and fix the remaining network parameters as follows. Firstly, we consider the subcritical regime ($\bar{\rho}_0 < \rho_{cr}$) in which ρ_{min} is constant, according to (4.22). On the other hand, Lemma 12 says that, for both (α, β, c) and $(\alpha, \beta, \bar{c})$-systems, there exists $c_1 > 0$ such that $T_{max} \leq c_1(\bar{\rho}_0 - \rho_{cr})^{-\gamma/2}$ for any large $\bar{\rho}_0$. Plugging this inequality to the right side of (4.37) obtains

$$\Psi_n \leq (1 + \rho_{min} - \bar{\rho}_0)c_1 x_m^{\gamma}(1-\alpha)^{\gamma/2}(\rho_{cr} - \bar{\rho}_0)^{\gamma/2} - (1 - \bar{\rho}_0),$$

which can be reduced to the second line of (4.50) by discarding the constant factors $(1 + \rho_{min})$ and $c_1 x_m^{\gamma}(1-\alpha)^{\gamma/2}$, and then applying Definition 19.

Secondly, we consider the supercritical regime ($\bar{\rho}_0 \geq \rho_{cr}$) for (α, β, c)-system. According to Lemma 12, there exist $c_2, \bar{\rho}_{0,3} > 0$ in that $\tau_{max} \leq c_3 \frac{\log(N)}{\sqrt{\bar{\rho}_0 - \rho_{cr}}}$ for any $\bar{\rho}_0 \geq \bar{\rho}_{0,3}$. This can be substituted into the upper bound of (4.37) to get

$$\Psi_n \leq c_2 x_m^{\gamma} \left((1-\alpha)/\log(N_c)\right)^{\gamma/2} (1 + \rho_{min} - \bar{\rho}_0)(\bar{\rho}_0 - \rho_{cr})^{\gamma/2} - (1 - \bar{\rho}_0). \tag{4.52}$$

After dropping the constant terms and noticing that $(1-\bar{\rho}_0)$ grows slower than $-\bar{\rho}_0(\bar{\rho}_0 - \rho_{cr})^{\gamma/2}$, we obtain the first line of (4.50). On the other hand, we can also get $\Psi_n \leq c_2 x_m^{\gamma} \left((1-\alpha)/\log(\alpha N_c)\right)^{\gamma/2} (1 + \rho_{min} - \bar{\rho}_0)(\bar{\rho}_0 - \rho_{cr})^{\gamma/2} - (1 - \bar{\rho}_0)$ for $(\alpha, \beta, \bar{c})$-system, which can be reduced to the first line of (4.50).

Finally, to prove (4.51), we vary N_c and re-use the inequality in (4.52). Then, (4.51) can be obtained by noticing that $c_2 x_m^{\gamma}(1-\alpha)^{\gamma/2}(1+\rho_{min}-\bar{\rho}_0)(\bar{\rho}_0-\rho_{cr})^{\gamma/2}$ and $(1-\bar{\rho}_0)$ are constants with respect to N_c. □

Corollary 10. *In D2D-based MSN with exponential-distributed session times, the scaling order of the node resilience Ψ_n versus the initial survival probability $\bar{\rho}_0$ satisfies*

$$\Psi_n = \begin{cases} O\left(\bar{\rho}_0 \exp\{-k_1(\bar{\rho}_0-\rho_{cr})^{-1/2}\}\right), & \text{for } \bar{\rho}_0 \geq \rho_{cr}, \\ O\left(\bar{\rho}_0 \exp\{-k_2(\rho_{cr}-\bar{\rho}_0)^{-1/2}\}\right), & \text{for } \bar{\rho}_0 < \rho_{cr}, \end{cases} \tag{4.53}$$

where $k_2 = c_1\lambda(1-\alpha)^{-1/2}$ and $k_1 = k_2\log(N_c)$. On the other hand, the scaling order of Ψ_n versus the number of communication nodes N_c for the supercritical regime ($\bar{\rho}_0 \geq \rho_{cr}$) satisfies

$$\Psi_n = O\left(-N_c^{-k_3}\exp\{-b_5 N_c\}\right), \tag{4.54}$$

where $k_3 = k_2(\bar{\rho}_0-\rho_{cr})^{1/2}$ is a constant.

Proof. Firstly, we consider the subcritical regime ($\bar{\rho}_0 < \rho_{cr}$) in which ρ_{min} is constant, according to (4.22). On the other hand, Lemma 12 says that, for both (α,β,c) and $(\alpha,\beta,\bar{c})$-systems, there exists $c_1 > 0$ such that $T_{max} \leq c_1(\bar{\rho}_0-\rho_{cr})^{-\gamma/2}$ for any large $\bar{\rho}_0$. Plugging this inequality to the right side of (4.44) obtains

$$\Psi_n \leq (1-\bar{\rho}_0) + (\bar{\rho}_0-\rho_{min})\exp\left\{-\frac{\lambda c_1}{(1-\alpha)^{1/2}(\bar{\rho}_0-\rho_{cr})^{1/2}}\right\},$$

which can be reduced to the second line of (4.53) by applying Definition 19.

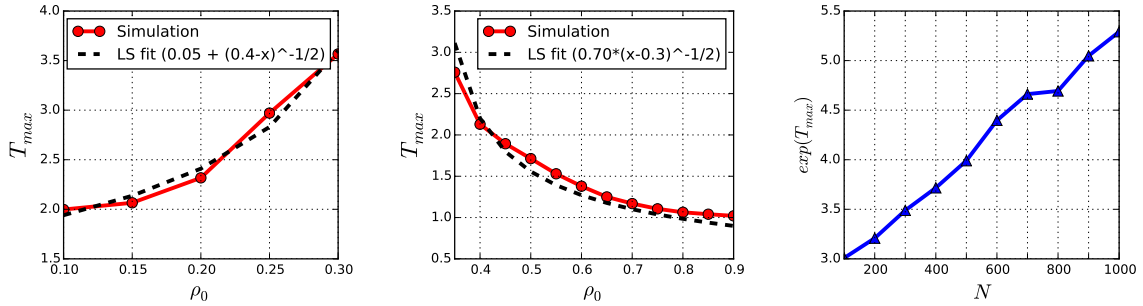
Secondly, we consider the supercritical regime ($\bar{\rho}_0 \geq \rho_{cr}$) for (α,β,c)-system. According to Lemma 12, there exist $c_2, \bar{\rho}_{0,3} > 0$ in that $\tau_{max} \leq c_3\frac{\log(N)}{\sqrt{\bar{\rho}_0-\rho_{cr}}}$ for any $\bar{\rho}_0 \geq \bar{\rho}_{0,3}$. This can be substituted into the upper bound of (4.37) to get

$$\Psi_n \leq (1-\bar{\rho}_0) + (\bar{\rho}_0-\rho_{min})\exp\left\{-\frac{\lambda c_1\log(N_c)}{(1-\alpha)^{1/2}(\bar{\rho}_0-\rho_{cr})^{1/2}}\right\},$$

After dropping the constant terms and noticing that $(1-\bar{\rho}_0)$ grows slower than $-\bar{\rho}_0(\bar{\rho}_0-\rho_{cr})^{\gamma/2}$, we obtain the first line of (4.50). We can get a similar inequality, also reducible to the first line of (4.50), for $(\alpha,\beta,\bar{c})$-system.

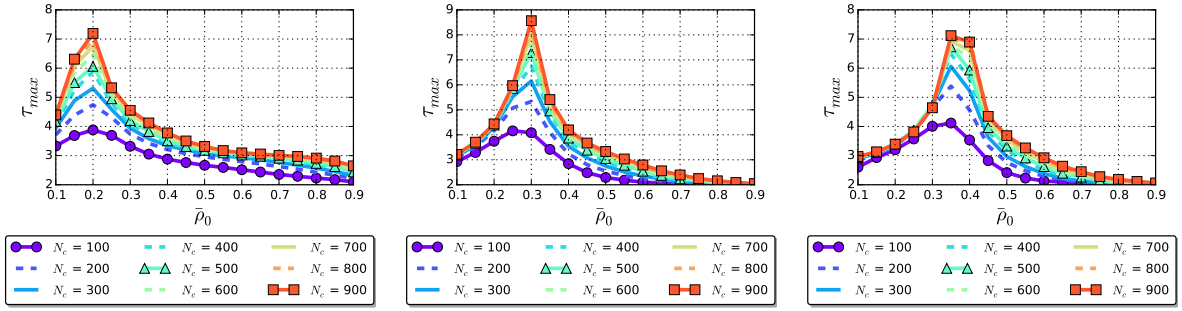
Finally, to prove (4.51), we vary N_c, employ the second line of (4.27), and re-use the inequality in (4.52) to get

$$\Psi_n \leq (1-\bar{\rho}_0 + c_4(-\exp\{-b_5 N_c\}))N_c^{-k_3}, \tag{4.55}$$



(a) Versus $\bar{\rho}_0$ in subcritical regime. (b) Versus $\bar{\rho}_0$ in supercritical regime.(c) Versus N_c in supercritical regime.

Figure 4.18: Scaling order of τ_{max} versus initial survival probability and number of nodes.



(a) $(\alpha, \beta, \bar{c})$-system. (b) $(\alpha, 0, c)$-system. (c) $(\alpha, 1, c)$-system.

Figure 4.19: Maximum deactivation time for various types of initial dropouts.

which can be simplified into (4.54) by dropping the constant $1 - \bar{\rho}_0$. □

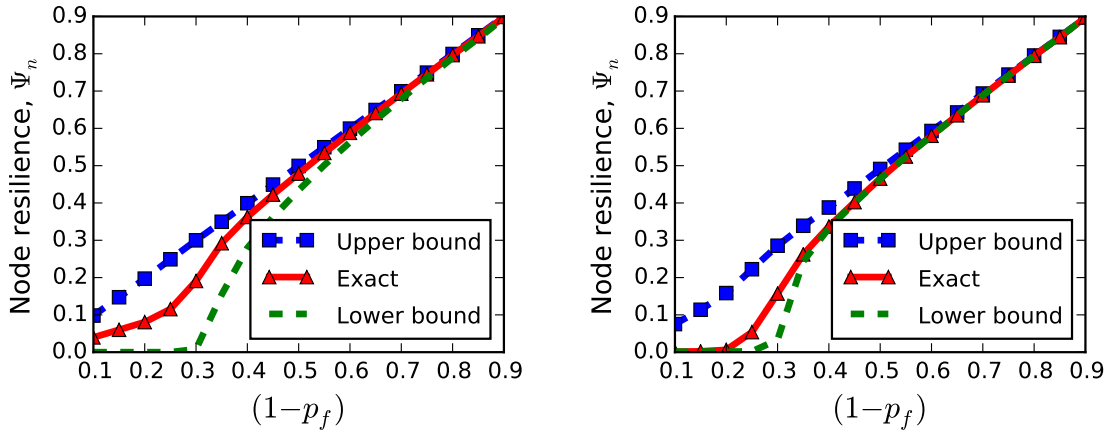
Note that the "Big-O" $O(\cdot)$ results in (4.50)-(4.51) and (4.53)-(4.54) only indicate the scaling order *upper bound* of Ψ_n. In the following, we will show that such an upper bound is *tight*.

4.7.4 Numerical Evaluation

After analyzing the bounds and scaling order of the node resilience metric, we validate our findings by using numerical simulations. Unless specified otherwise, the simulation parameters are identical to that in Sections 4.5.4 and 4.6.2. The results are presented as follows.

Evaluation of Maximum Deactivation Time

We start by examining the maximum deactivation time τ_{max}, which according to Theorem 8 and Corollary 9 is an important metric for analyzing the node resilience Ψ_n. To do this, we consider the $(\alpha, 0, c)$-system and plot τ_{max} versus the fraction of initial residues, $\bar{\rho}_0$, for the



(a) Exponential session times.

(b) Pareto session times.

Figure 4.20: Node resilience metric for $(\alpha, 0, c)$-system.

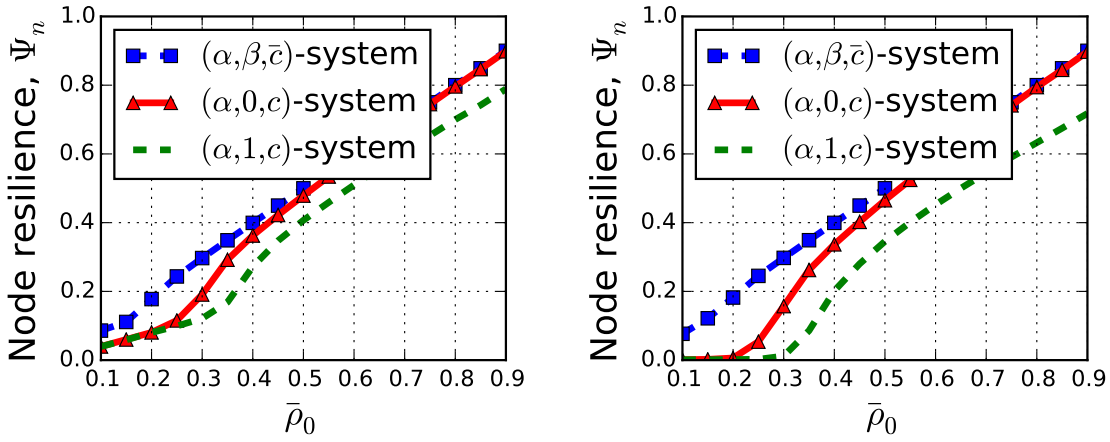
subcritical and supercritical regimes in Figs. 4.18a and 4.18b. We observe that the numerical results (the red solid plots) matches the scaling order analysis in Lemma 12 (see the black dotted lines for the least-square fit). More importantly, τ_{max} is a convex *increasing* function of $\bar{\rho}_0$ in the subcritical regime ($\bar{\rho}_0 < \rho_{cr}$) and a convex *decreasing* function when $\bar{\rho}_0 \geq \rho_{cr}$. This can also be observed for the $(\alpha, \beta, \bar{c})$ and $(\alpha, 1, c)$-systems with various numbers of communication nodes, N_c, in Figs. 4.19a-4.19c. Thus, determining the critical fraction of initial residues (i.e., $\rho_{cr} \approx \{0.2, 0.3, 0.35\}$ for $(\alpha, \beta, \bar{c})$, $(\alpha, 0, c)$, and $(\alpha, 1, c)$-systems, respectively) is equivalent to finding $\bar{\rho}_0$ that corresponds to the highest T_{max}.

Remark 14. *The fraction ρ_{cr} is known as a critical point, below which the network contraction process will disconnect the network completely [84]. Figures 4.18a, 4.18b, and 4.19 suggest that, in addition to solving (4.13), ρ_{cr} can be found as the $\bar{\rho}_0$ that leads to the highest τ_{max}. By comparing Figs. 4.11b and 4.19, we find an agreement between the resulting ρ_{cr} from both methods.*

Next, we consider the supercritical regime and verify the scaling order of τ_{max} versus N_c. To do this, we set $\bar{\rho}_0 = 0.65$ and plot $\exp(\tau_{max})$ versus N_c in Fig. 4.18c. The nearly-strait resulting plot suggests that the scaling order of $\Theta(\log(N_c))$ in Lemma 12 is valid. This means τ_{max} increases *sub-linearly* with respect to N_c and *the network contraction process propagates faster in large networks.*

Bounds of Node Resilience Metric

To evaluate the bounds of node resilience Ψ_n in (4.43) and (4.37), we set the mean session time to $E[ST_v] = 2$ time steps. For exponentially-distributed session times, we set $\lambda = 1/E[ST_u]$. On the



(a) Exponential session times.

(b) Pareto session times.

Figure 4.21: Node resilience metric for various systems.

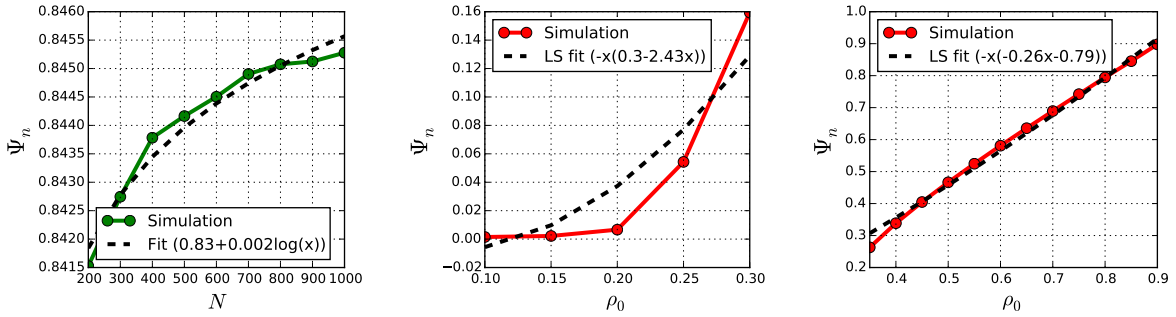
other hand, we fix $\gamma = 2$ to ensure that the mean is finite and assign $x_m = \frac{\gamma-1}{\gamma} E[ST_u] = 1$ to get the desired mean of $E[ST_u] = 2$ for Pareto-distributed session times. We plot the node resilience of the $(\alpha, 0, c)$-system with exponential and Pareto session time distributions in Figs. 4.20a and 4.20b, respectively. We observe that the numerical result (see the solid red plot) matches the lower and upper bounds (e.g., the dotted plots), thus validating Theorem 9 and Corollary 10. Moreover, unlike τ_{max} that exhibits a double scaling (i.e., τ_{max} increases and decreases below and above ρ_{cr}), the node resilience Ψ_n in Fig. 4.20 is strictly increasing with respect to $\bar{\rho}_0$, akin to the minimum utility ratio ρ_{min} in Fig. 4.11b. This indicates Ψ_n is dominated by ρ_{min}, as evidenced by its bounds in (4.43) and (4.37), which are linear functions of ρ_{min}.

Based on such a tightly-increasing relation between Ψ_n and ρ_{min}, we compare the node resilience of $(\alpha, \beta, \bar{c})$, $(\alpha, 0, c)$, and $(\alpha, 1, c)$-systems. According to the results in Figs. 4.10-4.14, we observe that $(\alpha, \beta, \bar{c})$-system generally achieves the highest ρ_{min}. This will translate to a large Ψ_n that is verified in Figs. 4.21, in which the $(\alpha, \beta, \bar{c})$-system outperform the others.

Remark 15. *The results in Fig. 4.21 indicate that end users' satisfaction, captured by Ψ_n, is most vulnerable to malware-induced initial dropouts (i.e., $(\alpha, 1, c)$-system). Thus, it is desirable to neutralize this threat by employing anti-malware schemes [125, 92], especially when social-induced, degree-weighted, and random communication-based initial dropouts are equally likely to occur.*

Scaling Order of Node Resilience Metric

We proceed by validating the scaling order of the node resilience Ψ_n in Theorem 9. We first focus on the result in (4.51) by considering an $(\alpha, 0, c)$-system with Pareto-distributed session



(a) Versus N_c in supercritical regime. (b) Versus $\bar{\rho}_0$ in subcritical regime. (c) Versus $\bar{\rho}_0$ in supercritical regime.

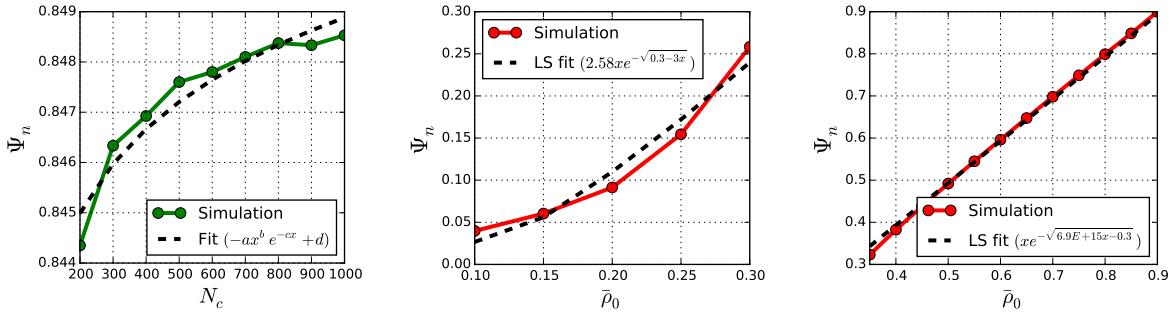
Figure 4.22: Scaling order of Ψ_n for Pareto-distributed session times.

times in the supercritical regime. To do this, we set $\bar{\rho}_0 = 0.85$ and plot Ψ_n versus the number of communication nodes, N_c, in Fig. 4.22a. Observe that the least-square fit of the scaling order in (4.51) matches with the numerical result, indicating that Ψ_n *increases logarithmically* with respect to N. Such an increase is because, according to Fig. 4.18c, the network contraction process will reach the last user at a later step when N_c is large, thus giving him/her more time to finish a session.

Next, we verify the scaling order of Ψ_n versus the fraction of initial residues $\bar{\rho}_0$ in (4.50). From the plots for both the subcritical and supercritical regimes of the $(\alpha, 0, c)$-system in Figs. 4.22b and 4.22b, we observe that the numerical and analytical results agree with each other, thus verifying Theorem 9. Moreover, unlike τ_{max} in Fig. 4.19 that exhibits a double scaling, Ψ_n is shown to *monotonically-increase* with respect to $\bar{\rho}_0$. As discussed in Section 4.7.4 above, this is because Ψ_n is dominated by $\bar{\rho}_0$. Specifically, users are less likely to be disconnected and can enjoy stable network connection when $\bar{\rho}_0$ is high.

Finally, by considering an $(\alpha, 0, c)$-system with exponential-distributed session times and then plotting the node resilience Ψ_n versus N_c and $\bar{\rho}_0$ in Figs. 4.23a-4.23c, we can also validate the scaling order in Corollary 10.

Remark 16. *The matching numerical and analytical results Figs. 4.22 and 4.23 indicate that the scaling order upper bound in Theorem 9 and Corollary 10 are tight, and Ψ_n can be enhanced by raising $\bar{\rho}_0$ (i.e., by installing anti-malware and/or using reliable hardware). In the supercritical regime, higher Ψ_n can be achieved by increasing N_c (i.e., allowing new nodes to join the D2D-based MSN).*



(a) Versus N_c in supercritical regime. (b) Versus $\bar{\rho}_0$ in subcritical regime. (c) Versus $\bar{\rho}_0$ in supercritical regime.

Figure 4.23: Scaling order of Ψ_n for exponential-distributed session times.

4.8 Summary

We studied how communication and social-based initial dropouts in D2D-based MSN may trigger a network contraction process, resulting in what we refer as (α, β, c) and $(\alpha, \beta, \bar{c})$-systems, respectively. To quantify the network-wide impact of initial dropouts, we proposed and derived a minimum utility ratio ρ_{min} in form of self-consistent equations, which gives an approximate solution to the numerical results using realistic D2D-based MSN trace. We show that the $(\alpha, \beta, \bar{c})$-system exhibits the best ρ_{min}, with up to 38% gap versus that of the (α, β, c)-system. Furthermore, we derived the scaling order of ρ_{min} with respect to the initial fraction of dropout nodes, average node degrees, social-to-communication dependency ratio, and number of nodes, and then present a simple method for detecting the root cause of initial dropouts. Based on the scaling order analysis, we proposed a `MinUtil` admission control and showed that it provides up to 36.74% improved average minimum utility ratio, than without admission. Finally, we derived the bounds and scaling order of a node resilience metric Ψ_n that quantifies the quality-of-experience from users' perspective. Through numerical evaluations, we verified that Ψ_n can be increased by suppressing the likelihood of initial dropouts and expanding the network size.

Chapter 5

Characterization of Edge-Induced Cascade-of-Failures in Cyber-Physical Networks

5.1 Background and Motivation

Cyber-physical system (CPS) refers to a collection of tightly-interacting and deeply-intertwined computational and physical nodes [164, 85], which has recently become an essential part of people's lives. Examples of CPS include communication-assisted power grid (i.e., smart grid), inter-vehicle communications, smart city monitoring, and Internet-of-Things [110]. In Fig. 5.1, we illustrate smart grid as a classic example of CPS, which consists of a power (physical) and a communication (cyber) networks. The power nodes provide electricity as the physical resource that the communication nodes depend upon. On the other hand, the communication nodes supply control signals that are critical to the power nodes' operation. As a result, CPS is usually modeled as two *interdependent* networks, i.e., cyber and physical networks.

In interdependent cyber-physical networks, reliability against faults is of a great concern. For example, it is critical to guarantee the correct operation of smart grid that provides energy to millions of its customers, especially under the current uprising of cyber attacks targeting power grids [95]. A small fault in cyber-physical systems can trigger the failure of successive parts, which is referred as a *cascade-of-failures* [107]. For instance, when node b_4 in Fig. 5.1 is faulty, its dependent nodes (e.g., c_4, c_5, c_7) are deactivated, which further causes the isolated nodes, c_1 to c_3, to also become deactivated. Thanks to the cyber-physical interdependence, the deactivations can rapidly spread, thus causing a network-wide cascade-of-failures. A cascade-of-failures will undermine the cyber-physical network's ability to supply resources to its customers. Not only the customers' experience will deteriorate, but also the operator's revenue will be diminished.

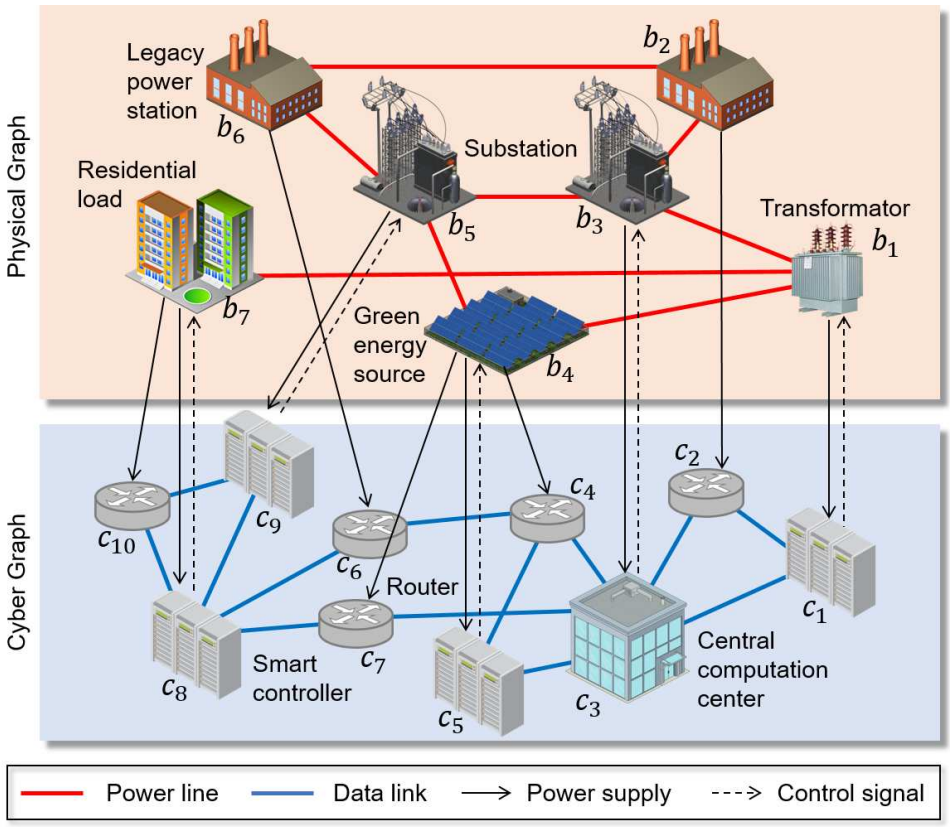


Figure 5.1: Smart grid as a collection of cyber and physical graphs.

Driven by the need to better understand how a cascade develops, many researchers have examined the behavior of cyber-physical networks under various types of faults. To this end, Buldyrev *et al.* [38] considered a simple interdependence model and proposed a percolation-based framework to quantify the occurrence likelihood of a giant component after a cascade. Yagan *et al.* [164] examined how regular allocation of interdependence links impacts the vulnerability of networks against failures. In [109], the most critical nodes that potentially cause large-scale cascades are identified using a greedy heuristic algorithm. Huang *et al.* examined the cyber-physical networks' connectivity under targeted attack rather than random faults [84]. Unfortunately, the existing researches are mainly focused on initial faults occurring at nodes, instead of edges.

Unlike nodes in cyber-physical networks that are highly-protected, the edges are typically unguarded and easier to break down. For example, power lines are open and exposed to the elements, in contrast to power stations that are protected by 24-hour surveillance. Hence, several scientists have attempted to study the impacts of edge disconnections to network connectivity. Specifically, Xiao and Yeh [160] had characterized the impact of cascading link failures by

modeling the power grid as a random geometric graph. Zhang *et al.* [172] had studied cascading failures due to power flow redistribution triggered by faulty power lines. These studies, however, only consider the power (physical) network while neglecting the cyber-physical interdependence. As a result, the impact of edge breakdown-induced cascade-of-failures in interdependent cyber-physical networks is still a wide open problem.

To study the aforementioned open problem, we present an analytical framework to model how edge fault-induced cascade-of-failures progresses. In particular, we consider two types of faults: when physical edges are disconnected (i.e., `Type-1`) and when both edges and cyber nodes break down (i.e., `Type-2`). Initial faults, which occur at time $t = 0$, cause nodes in the physical domain to fail. We model the set of non-failure physical nodes that remain functional as a time-decreasing process $\{\mathcal{P}_t : t \geq 0\}$. Each physical node excluded from $\{\mathcal{P}_t : t \geq 0\}$ triggers another process in the cyber domain that will eventually be acted back upon the physical domain, thus causing a cascade-of-failures. We characterize the influence of the cascade by measuring the fraction of physical nodes that remain functional eventually after the cascade-of-failures, which is referred as *node yield* Y_n. Then, we use mathematical analysis and numerical evaluation to understand how Y_n is affected by the cyber-physical networks' structure and the amount of edges initially at fault. Our findings and contributions can be summarized as follows.

1. We present a *two-step* framework: We first apply a mapping-based approach for transforming `Type-1` and `Type-2` faults in the original cyber-physical networks into a simpler node breakdown problem in equivalent networks. Then, percolation theory [38] is leveraged to characterize the interaction between the cyber and physical domains during a cascade-of-failures.

2. Using the proposed framework, we derive Y_n as a function of node degree distributions in the cyber and physical networks. By comparing our results to the existing results [38, 164, 109, 84], we show that Y_n under `Type-1` and `Type-2` faults is higher than a similar metric under node breakdowns. Hence, guarding against edge faults becomes less important than protecting nodes against initial breakdowns.

3. We further analyze the fraction of initial faults that will result in total network failure (e.g., $Y_n = 0$), referred as the critical disconnection ratio DR_{cr}. Under the special case of cyber-physical networks with Poisson-distributed degrees, we show that DR_{cr} increases sublinearly with the mean degree of the physical graph. Moreover, DR_{cr} can be increased by guarding cyber nodes against faults.

4. Finally, we propose a cascade length metric $\bar{\tau}_{CF}$ to quantify the amount of available time until a cascade-of-failures reaches all the failed nodes. Using scaling order analysis, we show

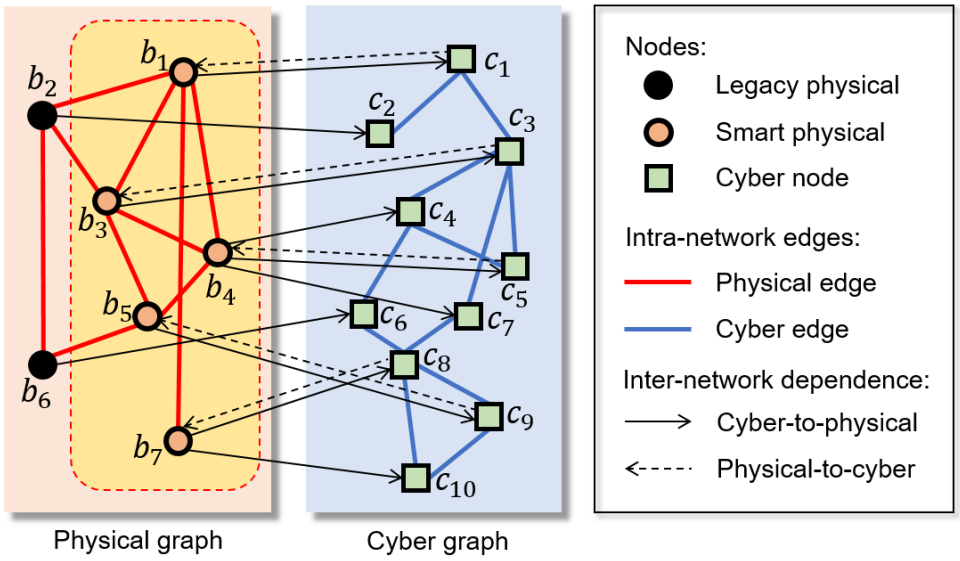


Figure 5.2: Illustration of cyber and physical graphs.

than $\bar{\tau}_{CF}$ is a non-monotone, increasing-then-decreasing function of the disconnection ratio and the maximum $\bar{\tau}_{CF}$ is achieved at the critical disconnection ratio DR_{cr}.

The remainder of this paper is organized as follows. In Section 5.2, we formalize the cyber-physical networks and node yield Y_n as our metric-of-interest. In Section 5.3, we analyze Y_n via mathematical and numerical evaluations. In Section 5.4, the critical disconnection ratio DR_{cr} is studied. Finally, the paper is concluded in Section 5.6.

5.2 Models and Problem Formulation

In this section, we define the models of cyber-physical networks, initial faults, and cascade-of-failures. Then, we state our main research problem.

5.2.1 Model of Interdependent Cyber-Physical Networks

We first study the cyber and physical graphs, and then explain how their interdependence can be modeled.

Cyber and Physical Graphs

We consider a collection of co-existing networks, which is abstracted as a pair of cyber and physical graphs defined as follows. Note that the collections of nodes and edges in a graph also forms a network; thus, we use the terms "graph" and "network" interchangeably.

Definition 23. *Let $\mathcal{P} = \{b_1, \ldots, b_{n_p}\}$ be the set of physical nodes, and $\mathcal{E}_p$ be the edges between the physical nodes. Then, physical graph is defined as $\mathcal{G}_p = (\mathcal{P}, \mathcal{E}_p)$. On the other hand, $\mathcal{G}_c = (\mathcal{C}, \mathcal{E}_c)$ is the cyber graph, where $\mathcal{C} = \{c_1, \ldots, c_{n_c}\}$ and $\mathcal{E}_c$ are the cyber nodes and edges, respectively.*

We present in Fig. 5.2 an example of the physical and cyber graphs. In Fig. 5.2, physical edges (the red links) enable physical resources—such as electricity, water, oil, and gas—to be exchanged between the physical nodes (see the circles on the left). There are two types of physical nodes. The first one is the so-called *smart nodes* (i.e., b_1, b_3-b_5, and b_7) that can autonomously cooperate with each other, to support a robust and efficient network operation. The remaining nodes (i.e., b_2 and b_6), on the other hand, are *legacy* (dumb) nodes that must be controlled manually. The number of the former smart nodes is proportional to the following ratio.

Definition 24 (Adoption ratio)**.** *Let $\mathcal{P}_s \subset \mathcal{P}$ be the set of smart physical nodes. Then, adoption ratio is defined as $\alpha := \frac{|\mathcal{P}_s|}{|\mathcal{P}|}$.*

The adoption ratio captures the progressive adaptation phase of smart physical nodes, from the beginning of their implementation (when $\alpha \approx 0$) until they are fully adopted ($\alpha = 1$).

In addition to the physical nodes and edges, Fig. 5.1 also consists of a cyber graph $\mathcal{G}_c$ (see the blue highlighted area). To this end, nodes in $\mathcal{G}_c$ indicate the logical endpoints of information, whose interconnection is described by the cyber edges (see the blue lines). Together, the cyber nodes and edges are capable of providing information, or even regulate the operation of the physical graph [148].

Interdependence Between Cyber and Physical Graphs

We assume that the aforementioned cyber and physical graphs are coupled by a two-way relationship as follows.

Definition 25 (Interdependence)**.** *Graphs $\mathcal{G}_a$ and $\mathcal{G}_b$ are said to be interdependent if they have the following relationships:*

- *Cyber-to-physical dependence $\mathcal{E}_{c\to p} := \{e(u \to v) : u \in \mathcal{P}, v \in \mathcal{C}\}$, where edge $e(u \to v)$ indicates that node u provides physical resource to a cyber node v, which will cease to function without the resource provided by u.*
- *Physical-to-cyber dependence $\mathcal{E}_{c\leftarrow p} := \{e(u \leftarrow v) : u \in \mathcal{P}, v \in \mathcal{C}\}$, where $e(u \leftarrow v)$ indicates that v provides cyber resource to a physical node u and u cannot function without the resource provided by v.*

The cyber-to-physical dependence $\mathcal{E}_{c\to p}$ is motivated by practical cyber nodes, such as routers and smart controllers, that depend on the physical resources (e.g., electricity) provided by the physical nodes. To this end, we assume a *many-to-one dependence*: each cyber node requires resource from exactly one physical node $u \in \mathcal{P}$, while a node u can provide resources to multiple cyber nodes (see the solid horizontal arrows in Fig. 5.2). The many-to-one dependence can be modeled using a *balls-and-bins* problem [43, Sec. 3.2] as follows: every ball (e.g., cyber node) is assigned randomly to the available bins (buses). Denote $n_p = |\mathcal{P}|$ and $n_c = |\mathcal{C}|$ as the number of physical and cyber nodes, respectively. Let $\mathcal{D}(u) \subset \mathcal{C}$ be the set of communication (cyber) nodes that depend on the resource from node u, and $dep(u) := |\mathcal{D}(u)|$ be the number of dependent nodes. Then, the likelihood that node u provides resources to k cyber nodes is given as

$$P(k) := \Pr\{dep(u) = k\} = \binom{n_c}{k} n_p^{-k} \left(1 - \frac{1}{n_p}\right)^{n_c - k}, \tag{5.1}$$

which is a Binomial distribution with parameters $(n_c, \frac{1}{n_p})$ [43].

The physical-to-cyber dependence $\mathcal{E}_{p\leftarrow c}$, on the other hand, is motivated by the fact that the cyber network must provide the information and control signals to support the smart nodes' operation. We assume a *one-to-one* physical-to-cyber dependence as follows: each smart node $u \in \mathcal{P}_s$ depends on one cyber node, which is selected uniformly at random from the set of dependent nodes, $\mathcal{D}(u)$. From the cyber graph's perspective, the probability that node $v \in \mathcal{C}$ has a physical-to-cyber edge $e(u \leftarrow v) \in \mathcal{E}_{p\leftarrow c}$ is equal to

$$\sum_{i=1}^{|\mathcal{P}_s|} \sum_{k=1}^{n_c} \Pr\{dep_u = k\} \frac{k}{n_c} \times \frac{1}{k} = \frac{|\mathcal{P}_s|}{n_c}. \tag{5.2}$$

As illustrated in Fig. 5.2, edges $\mathcal{E}_{p\leftarrow c}$ becomes a subset of the cyber-to-physical edges. We refer the cyber nodes with both types of edges, such as $\{c_1, c_3, c_5, c_8, c_9\}$ in Fig 5.1, to have a *bi-directional* dependence. Together with the cyber and physical graphs ($\mathcal{G}_p$ and $\mathcal{G}_c$), the dependence edges $\mathcal{E}_{p\to c}$ and $\mathcal{E}_{p\leftarrow c}$ form a set of *interdependent cyber-physical networks*.

5.2.2 Initial Faults in Interdependent Cyber-Physical Networks

Let the lifetime of the interdependent cyber-physical networks be divided into discrete and non-negative time steps $\{t \geq 0\}$. We assume that at the initial time $t = 0$, the interdependent networks are impaired by the following faults.

Type-1 (Initial faults at physical edges)

Physical edges such as power and oil lines can be damaged by man-made sabotages and natural disasters. To model such a physical edge disconnection, we define the following metrics.

Definition 26 (Edge ratios)**.** *Let $\mathcal{E}_{fault} \subset \mathcal{E}_p$ be the initial set of disconnected physical edges. The edge disconnection ratio is defined as $DR := E\left[\frac{|\mathcal{E}_{fault}|}{|\mathcal{E}_p|}\right]$, while edge survival ratio is the fraction $\phi = 1 - DR$.*

Under the so-called `Type-1` initial fault, every physical edge becomes disconnected with probability $DR = 1 - \phi$.

Type-2 (Initial faults at physical edges and cyber nodes)

In addition to the initial edge faults, the cyber nodes may also be brought down by denial-of-service (DoS) attacks [148] and malicious software infection [108]. Although node break down in cyber-physical networks has been widely studied [38, 109, 84], the damage of its occurrence together with edge disconnections remains an open problem. To model such a *joint edge and node* break downs, we further present the following metrics.

Definition 27 (Node ratios)**.** *Let $\mathcal{C}_{fault} \subset \mathcal{C}$ be the initial set of faulty cyber nodes. Given a node-to-line ratio $\kappa := E\left[\frac{|\mathcal{C}_{fault}|}{|\mathcal{E}_{fault}|}\right]$, the node disconnection ratio is defined as $DR_n := \kappa \times DR$.*

Under what is referred as a `Type-2` scenario, we assume that initial faults occur uniformly at random on both: (i) the physical edges, with probability DR, and (ii) the cyber nodes, with likelihood DR_n.

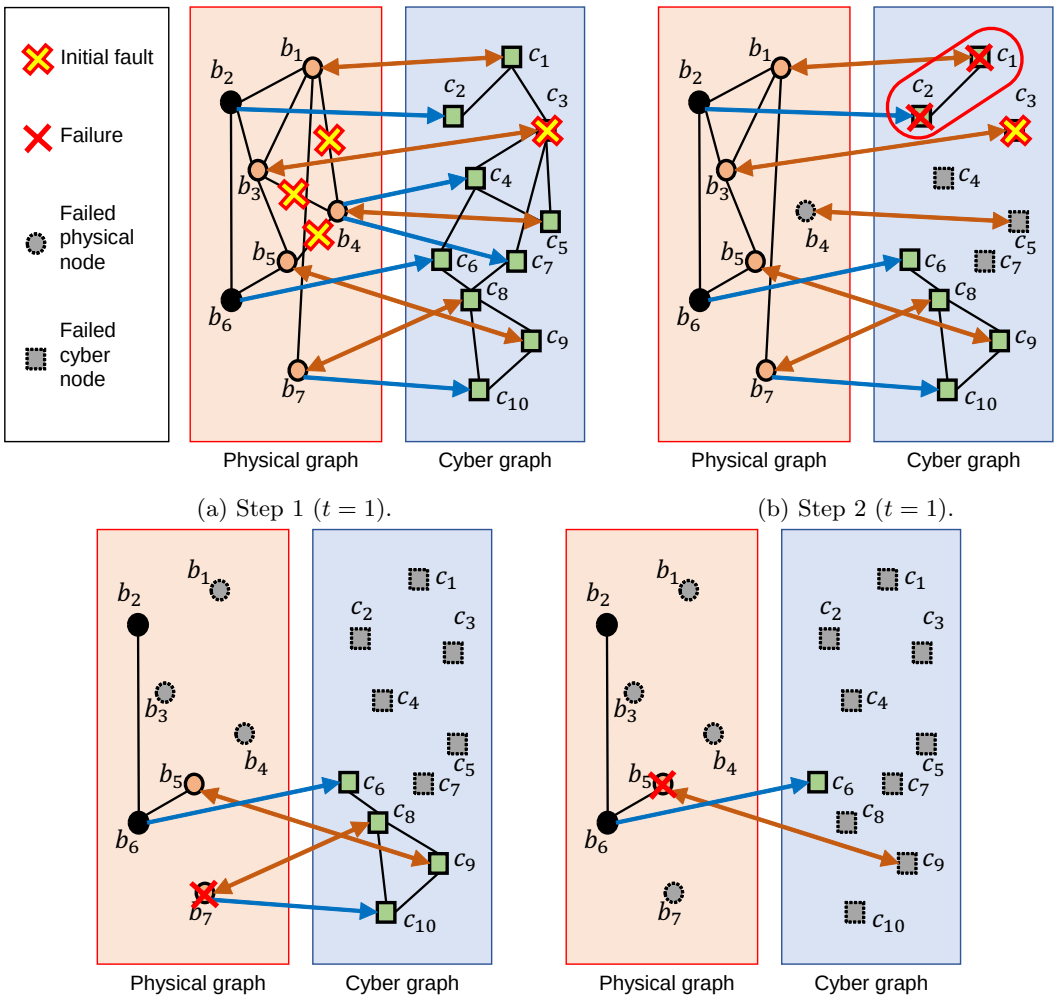
5.2.3 Fault-Induced Cascade-of-Failures

After discussing about `Type-1` and `Type-2` faults at time origin ($t = 0$), we describe how they could trigger *failures*, that is, the deactivations of more nodes at subsequent time steps ($t \geq 1$). Specifically, we focus on *cascading failures*, i.e., the set of failed nodes that grows larger over time.

To study how cascading failures are formed, we employ the example in Fig. 5.3 where we focus on the `Type-2` initial faults.[1] Then, we consider the following steps.

Step 1: In our example, cyber node $\mathcal{C}_{fault} = \{c_3\}$ and physical edges $\mathcal{E}_{fault} = \{e(b_1, b_4), e(b_3, b_4), e(b_4, b_5)\}$ are initially at fault (see Fig. 5.3a). We first deal with the faulty edges, while the contribution of $\mathcal{C}_{fault}$ will be accounted for in Step 2 below. To capture how edge disconnection affects the physical graph, we define the following ratio.

[1] Note that `Type-1` is a special case of `Type-2` fault with no initial fault at the cyber nodes (i.e., $\mathcal{C}_{fault} = \emptyset$).



(a) Step 1 ($t = 1$). (b) Step 2 ($t = 1$).

(c) Step 3 ($t = 2$). (d) Step 4 ($t = 2$).

Figure 5.3: Step-by-step illustration of cascading failures in cyber-physical networks.

Definition 28 (Physical utility ratio)**.** *Let $\mathcal{P}'_t \subset \mathcal{P}$ be the set of residual physical nodes at time t. Denote $\mathcal{P}_t \subset \mathcal{P}'_t$ as the set of residual physical nodes that stay functional. Then, physical utility ratio is defined as $x_t := \frac{|\mathcal{P}_t|}{|\mathcal{P}|}$.*

According to our example in Fig. 5.3a, after the faulty edges $\mathcal{E}_{fault}$ have been removed, b_4 becomes disconnected from the rest of the physical nodes. As a result, the set of remaining nodes in the first step is given as $\mathcal{P}'_1 = \mathcal{P} \backslash \{b_4\}$. To account for the worst-case impact, we assume that only physical nodes belonging to the largest connected component (LCC) of $\mathcal{G}_p(\mathcal{P}'_t)$ remain functional, where $\mathcal{G}(\mathcal{V})$ is a vertex-induced subgraph of $\mathcal{G}$ [154]. Such an LCC-based assumption simplifies analysis and has been widely used in network theory [38, 85, 164]. Since in our example all the nodes still belong to the LCC, we have $\mathcal{P}_1 = \mathcal{P}'_1$ and a physical utility ratio of $x_1 = \frac{6}{7}$.

To this end, physical utility ratio captures the fraction of physical nodes that remain functional and can still be utilized up to time $t = 1$.

Step 2: Next, we consider the cyber-to-physical dependence $\mathcal{E}_{p\to c}$ and examine its impact to the connectivity of the cyber graph. To this end, in addition to the initial fault at $\mathcal{C}_{fault} = \{c_3\}$, cyber nodes $\{c_4, c_5, c_7\}$ are also deactivated due to the cyber-to-physical dependence.

Definition 29 (Cyber utility ratio)**.** *Let $\mathcal{C}'_t \subset \mathcal{C}$ be the set of residual cyber nodes, and $\mathcal{C}_t \subset \mathcal{C}'_t$ be the set of functional cyber nodes. Then, cyber utility ratio is defined as $y_t := \frac{|\mathcal{C}_t|}{|\mathcal{C}|}$.*

After the deactivation of nodes $\mathcal{C}_{fault} \cup \{c_4, c_5, c_7\}$, the set of residual cyber nodes becomes $\mathcal{C}'_1 = \{c_1, c_2, c_6, c_8, \ldots, c_{10}\}$. According to Fig. 5.3b, the residual nodes are further separated into two components: an LCC containing $\mathcal{C}'_1 \setminus \{c_1, c_2\}$ and a smaller one containing the remaining nodes. Similar to the previous step for physical graph, we assume only cyber nodes belonging to the LCC remain functional. As a result, nodes c_1 and c_2 are isolated and deactivated, the set of functional nodes becomes $\mathcal{C}_1 = \mathcal{C}'_1 \setminus \{c_1, c_2\}$, and the cyber utility ratio is given as $y_1 = 0.5$.

Step 3: We increment the time to $t = 2$, take into account the physical-to-cyber dependence $\mathcal{E}_{p\leftarrow c}$, and re-examine the physical graph's structure. To this end, physical nodes b_1 and b_3, which depend on the non-functional cyber nodes c_1 and c_3, also become deactivated. Among the residual physical nodes $\mathcal{P}'_2 = \mathcal{P}_1 \setminus \{b_1, b_3\}$, node b_7 is isolated from the rest and becomes non-functional. As a result, the set of functional nodes becomes $\mathcal{P}_2 = \mathcal{P}'_2 \setminus \{b_7\}$ and the physical utility ratio is now given as $x_2 = \frac{3}{7}$.

Steps 4 and beyond: By continuing the steps above and incrementing time t at every odd step, we find that all the cyber nodes will eventually be disconnected completely (i.e., $\mathcal{C}_t = \emptyset$ and $y_t = 0$ for any $t \geq 3$), while only the legacy physical nodes (b_2 and b_6) remain functional. This step-by-step example illustrates how small initial faults at nodes and edges ($\mathcal{C}_{fault}$ and $\mathcal{E}_{fault}$) may trigger a network-wide *cascade-of-failures*, with only $x_t = \frac{2}{7}$ physical utility ratio after the cascade ends.

5.2.4 Problem Formulation

In cyber-physical networks, the physical utility ratio x_t is proportional to the amount of functional nodes that still can supply physical resources, even after a cascade-of-failures. The physical resources are vital not only to the cyber nodes, but also for customers supported by the networks. For example, in smart grid systems, higher x_t means a larger electricity demand can be served by the networks, which translates into better customer satisfaction and larger operator revenue. Thus, it is of high interest to quantify x_t.

Since the functional nodes $\mathcal{P}_t$ is a decreasing set (e.g., $\mathcal{P}_t \subseteq \mathcal{P}_{t-1}$ for all t), then the corresponding physical utility ratio $x_t = \frac{|\mathcal{P}_t|}{|\mathcal{P}|}$ is also declining over time. Moreover, x_t is random

variable since it depends on the set of initial faults, $\mathcal{E}_{fault}$ and $\mathcal{C}_{fault}$, which result from random selection processes. Therefore, x_t is a random yet decreasing number, and we define the following metric to quantify x_t.

Definition 30 (Node yield)**.** *Let $E(\cdot)$ be an expectation operator. Then, node yield is defined as*

$$Y_n := \min_{t \geq 0} E[x_t] = \lim_{t \to \infty} E[x_t]. \tag{5.3}$$

A higher node yield means that under fault-induced cascade-of failures, the cyber-physical networks are more likely to provide sufficient resources to their customers. Otherwise, lower Y_n indicates a more extensive damage.

In this paper, we quantify the extensive impact brought by the cascade-of-failures to the cyber-physical networks. Specifically, we ask, *what is the node yield of interdependent cyber-physical networks with `Type-1` or `2` initial faults?*

5.2.5 Use Case Examples

In previous subsections, we have presented generic definitions and formulations regarding cascade-of-failures in cyber-physical networks. Before proceeding to analysis, we outline several practical cases that will benefit from our formulation.

Case 1: Smart grid

One of the most popular examples of cyber-physical networks is communication-assisted power grid—also known as a *smart grid* [137, 151], which is depicted in Fig. 5.1. In our illustration, the physical graph $\mathcal{G}_p$ of a smart grid comprises a collection of nodes, which represent the points where electricity is generated (b_2, b_4, and b_6), distributed (b_1, b_3, and b_5), or consumed (b_7). The physical graph $\mathcal{G}_p$ also consists of edges (see the red lines) that serve as a medium for transferring electricity between buses.

On the other hand, the cyber graph $\mathcal{G}_c$ consists of computation stations, smart controller, and data router as the nodes. With the aid of communication edges (see the blue lines in Fig. 5.1), the cyber nodes not only generate, forward, and/or consume data, but also provide necessary signals to regulate the power grid's (physical network's) operation.

In smart grid, faults may occur due to the following reasons.

1. `Type-1` **- Initial faults at physical edges:** Physical edges are typically installed as overhead high-voltage lines, which are readily exposed to the elements of nature. Thus, natural disasters like tornadoes, hurricanes, and earthquakes can topple down the exposed

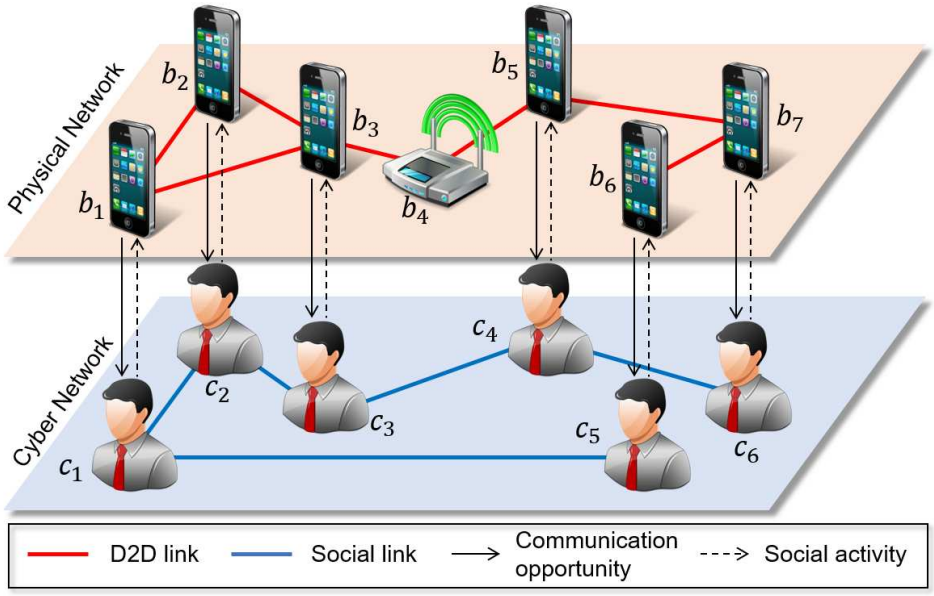


Figure 5.4: D2D-based MSN as a collection of cyber and physical graphs.

power lines. In fact, disasters are among the major causes of electricity blackouts, as exemplified by the massive blizzard that hits millions of Americans in January 2016 [18].

2. `Type-2` **- Initial faults at physical edges and cyber nodes:** A coronal mass ejection (CME) is a natural release of the sun's plasma, which further sends a solar storm to earth's surface [64]. Solar storm causes magnetic field variations that may increase the electrical current flowing through smart grid's power lines. As a result, the overloaded line heats up and sags onto the ground, triggering a short circuit that disconnects the lines. In addition, the geomagnetic fluctuations caused by CME may shut down sensitive communication equipment, causing faults at the cyber nodes.

Remark 17. *In a smart grid, node yield Y_n is different but related to demand yield Y [28], which represents the amount of serviceable electricity demand after cascades, divided by the original demand. Larger Y_n indicates a higher likelihood to find additional generator nodes for serving demands. Thus, node yield Y_n becomes proportional to the demand yield.*

Case 2: D2D-based MSN

Device-to-device (D2D)-based mobile social networking (MSN) is a new paradigm that enable proximate users to form ad-hoc networks for exchanging multimedia contents. Some popular D2D-based MSNs, including FireChat and Jott, have attracted attention from million of users [51, 8]. As indicated in Fig. 5.4, a D2D-based MSN consists of wireless devices, such as smart

phones and tablets, as well as short range D2D links between the devices. Together, the devices and links create a physical graph $\mathcal{G}_p$ (see the red shaded area).

A D2D-based MSN also has a cyber graph $\mathcal{G}_c$ (see the blue shaded area on the bottom) that consists of MSN users and their social links. To this end, the social links indicate whether two users share a common interest (e.g., belonging to the same chat room), while MSN users are the endpoints of social contents. For instance, a content may have c_1 and c_5 as its source and destination, respectively. The content, on the other hand, is delivered as packets through the physical graph (i.e., from nodes b_1 to b_6 via links (edges) $e(b_1, b_3)$, $e(b_3, b_4)$, $e(b_4, b_5)$, $e(b_5, b_6)$, and $e(b_6, b_7)$).

In a D2D-based MSN, the cyber-to-physical dependence comes from the social users that require D2D (physical) connectivity for exchanging social contents. On the other hand, the D2D wireless devices also rely on the social users that may disable their devices when becoming socially inactive. As a result, D2D-based MSN can be modeled as a special case of interdependent networks (see Definition 25) when $|\mathcal{C}| = |\mathcal{P}_s|$, all the physical resources are provided by the smart nodes $\mathcal{P}_s$, and each of the dependence edge is bi-directional.

Among the initial faults that may occur in D2D-based MSN, we consider the following two types.

1. `Type-1` **- Initial faults at physical edges:** Two wireless nodes may be separated by human bodies and other objects, which causes severe attenuation to D2D communications with frequency 1GHz and beyond [153]. As a result, wireless links (edges) may experience disconnections, especially under dynamic environments such as road-side setting and live concert.

2. `Type-2` **- Initial faults at physical edges and cyber nodes:** In addition to intermittent edge connectivity due to environment-induced attenuation, social users (cyber nodes) may tend to become inactive over time [147]. From the cyber network's perspective, inactive nodes can be viewed to be dropped out, thus causing a `Type-2` fault.

5.3 Analysis of Node Yield

In this section, we mathematically analyze the node yield under `Type-1` and `Type-2` initial faults. Our analysis not only serves as a tool for predicting the coverage of fault-induced cascade-of-failures, but also provides guidelines on how to defend cyber-physical networks against threats.

5.3.1 Node Yield Under Type-1 Initial Faults

In this subsection, we will show how to analyze the node yield Y_n, given the edge disconnection ratio DR and the statistical structure of the cyber and physical graphs, $\mathcal{G}_c$ and $\mathcal{G}_p$. Unlike the existing researches that mainly consider the impact of cascading failures triggered by initial faults at nodes [38, 84, 85], we focus on characterizing the impacts of initial faults at edges (`Type-1`), which remains an open yet challenging problem. To tackle our research problem, we adapt the *mapping-based* approach, which was first applied to transform a *targeted node attack* problem[2] in interdependent networks into a *random node attack*[2] in a transformed pair of interdependent networks [84]. By adapting the mapping-based framework, our problem can be re-stated as follows.

- Find an auxiliary physical graph $\tilde{\mathcal{G}}_p$, so that
- a `Type-1` fault in interdependent networks $(\mathcal{G}_p, \mathcal{G}_c)$ can be viewed as a "random node attack" in $(\tilde{\mathcal{G}}_p, \mathcal{G}_c)$.

In the following, we first deal with the latter, that is, to analyze node yield in the transformed networks $(\tilde{\mathcal{G}}_p, \mathcal{G}_c)$ under a random node attack scenario. Then, we proceed by tackling the former part, i.e., to transform the statistical property of the original graph $\mathcal{G}_p$ to that of an auxiliary graph $\tilde{\mathcal{G}}_p$.

Node Yield in Transformed Interdependent Networks

We consider an auxiliary physical graph $\tilde{\mathcal{G}}_p$ and a cyber graph $\mathcal{G}_c$ with initial faults occurring randomly at the physical nodes with probability $(1-p_p)$, which we simply refer as `Type-0` fault. Note that p_p is an auxiliary variable and its relation to the edge disconnection ratio DR will be derived in Section 5.3.1 below. Under `Type-0` fault, the considered networks looks similar to the cyber-physical networks with one-to-multiple dependence in [85]. Compared to [85], our networks ($\tilde{\mathcal{G}}_p$ and $\mathcal{G}_s$) are differs in that only an $\alpha \in (0, 1]$ fraction of the physical nodes are smart and depend on the cyber nodes. Hence, the existing result for networks with entirely smart physical nodes ($\alpha = 1$) [85] cannot be applied directly to our problem.

To proceed, we focus on the auxiliary physical graph $\tilde{\mathcal{G}}_p = (\tilde{\mathcal{P}}, \tilde{\mathcal{E}}_p)$ and denote the degree distribution of $\tilde{\mathcal{G}}_p$ as

$$\tilde{P}_p(k) := Pr\{|\mathcal{N}_p(u)| = k\}, \tag{5.4}$$

where $\mathcal{N}_p(u) := \{e(u, v) \in \tilde{\mathcal{E}}_p : u \in \tilde{\mathcal{P}}\}$ is the set of node u's neighbors. When a $(1-p_p)$ fraction of physical nodes are removed from $\tilde{\mathcal{G}}_p$, the residual nodes may become fragmented into several

[2] *Random node attack* refers to the case when initial faults occur at nodes, with uniform likelihood. *Targeted node attack*, on the other hand, concentrates the faults on popular nodes with many neighbors.

components. Among the residues, the fraction of nodes that still belong to the LCC of $\tilde{\mathcal{G}}_p$ is given as

$$\tilde{g}_p(p_p) = 1 - \tilde{G}_{p,1}(1 - p_p(1 - f_p)), \tag{5.5}$$

where $\tilde{G}_{p,1}(x) := \frac{\tilde{G}'_{p,0}(x)}{\tilde{G}'_{p,0}(1)}$ is a supplementary function, $\tilde{G}'_{p,0}(x) := \frac{d}{dx}\tilde{G}_{p,0}(x)$ denotes a first-order derivative, and

$$\tilde{G}_{p,0}(x) := \sum_{k=0}^{\infty} \tilde{P}_p(k)x^k \tag{5.6}$$

is the probability generating function of the degree distribution. Moreover, f_p is a variable that satisfies a transcendental (self-consistent) equation

$$f_p = \tilde{G}_{p,1}(1 - p_p(1 - f_p)). \tag{5.7}$$

Similar denotations also exist for the cyber network $\mathcal{G}_c$, by substituting the subscript $()_p$ with $()_c$ and omitting the tilde sign from $\tilde{\mathcal{G}}$ and $\tilde{P}$. Note that the fractions $\tilde{g}_p$ and g_s will be useful in analyzing the expected cyber and physical utility ratios in the following lemma.

Lemma 13 (Expected utility ratios)**.** *Denote $\bar{x}_t := E[x_t]$ and $\bar{y}_t := E[y_t]$ as the expected physical and cyber utility ratios at time t. Then, under* `Type-1` *initial fault,*

$$\bar{x}_t = \bar{x}'_t \times \tilde{g}_p(\bar{x}'_t), \text{ and} \tag{5.8}$$

$$\bar{y}_t = \bar{y}'_t \times g_c(\bar{y}'_t), \tag{5.9}$$

where $\bar{x}'_t := E[\frac{|\mathcal{P}'_t|}{|\mathcal{P}|}]$ and $\bar{y}'_t := E[\frac{|\mathcal{C}'_t|}{|\mathcal{C}|}]$ are the mean fraction of physical and cyber residues, respectively, that satisfy

$$\begin{cases} \bar{x}'_t &= p_p \left[1 - \alpha \left(1 - g_c(\bar{y}'_{t-1})\right)\right], \\ \bar{y}'_t &= p_p \times \tilde{g}_p(\bar{x}'_t). \end{cases} \tag{5.10}$$

Proof. The sequence of node failures of the first four time steps are outlined below. In our analysis, the numbers of nodes refer to their expected value; we omit the term "expected" for simplicity.

- **Step 1 at time $t = 1$:** After initially disconnecting $(1 - p_p)$ fraction of physical nodes from $\tilde{\mathcal{G}}_p$ at random, the number of residual physical nodes becomes[3]

$$E|\mathcal{P}'_1| = x'_1|\mathcal{P}| = p_p|\mathcal{P}|. \tag{5.11}$$

[3]We assume the numbers of physical and cyber edges are static, e.g., $E|\mathcal{P}| = |\mathcal{P}|$ and $E|\mathcal{C}| = |\mathcal{C}|$.

Then, the number of residual nodes that belong to the LCC of $\tilde{\mathcal{G}}_p$ and remain functional becomes

$$E|\mathcal{P}_1| = E|\mathcal{P}_1'|\tilde{g}_p(x_1') = p_p\tilde{g}_p(x_1')|\mathcal{P}| = x_1|\mathcal{P}|. \tag{5.12}$$

- **Step 2 at time $t = 1$:** Next, we take into account the social-to-physical dependence $\mathcal{E}_{p\to c}$ and focus on the number of residual social nodes. From the previous step, there are $E|\mathcal{P}_1|$ of functional physical nodes and the residual nodes must have their physical resources supplied by (e.g., be connected with $\mathcal{E}_{p\to c}$ edges to) nodes $\mathcal{P}_1$. Since each of the physical nodes in $\mathcal{P}_1$ has an average number of $E(K) = \sum_{k=1}^{n_c} kP(k) = \frac{|\mathcal{C}|}{|\mathcal{P}|}$ social-to-physical dependence edges, then the number of residual cyber nodes becomes

$$E|\mathcal{C}_1'| = E(K) \times E|\mathcal{P}_1| = E|\mathcal{P}_1|\frac{|\mathcal{C}|}{|\mathcal{P}|} = y_1'|\mathcal{C}|. \tag{5.13}$$

Among the residual cyber nodes, the number of functional nodes that still belong to the LCC is given as

$$E|\mathcal{C}_1| = E| \times \mathcal{C}_1'|g_c(y_1') = y_1' \times g_c(y_1')|\mathcal{C}| = y_1|\mathcal{C}|. \tag{5.14}$$

- **Step 3 at time $t = 2$:** We return to the structure of the physical graph $\mathcal{G}_p$. In $\mathcal{G}_p$, only an α portion of functional nodes $\mathcal{P}_1$ are smart nodes that have physical-to-cyber dependence $\mathcal{E}_{p\leftarrow c}$ and depend on the cyber nodes. As a result, the removal of an additional $1 - \frac{E|\mathcal{C}_1|}{E|\mathcal{C}_1'|}$ fraction of cyber nodes that do not belong to the LCC, in Step 2 above, will result in an additional failure to $\alpha \times \left(1 - \frac{E|\mathcal{C}_1|}{E|\mathcal{C}_1'|}\right)$ fraction of physical nodes with physical-to-cyber dependence. To this end, the number of remaining physical nodes can be expressed as

$$\begin{aligned} E|\mathcal{P}_2'| &= E|\mathcal{P}_1| - \alpha\left(1 - \frac{E|\mathcal{C}_1|}{E|\mathcal{C}_1'|}\right)E|\mathcal{P}_1| \\ &= \left(\alpha \times g_c(y_1') + (1-\alpha)\right)E|\mathcal{P}_1|, \end{aligned} \tag{5.15}$$

that is, from $\mathcal{P}_1$ to $\mathcal{P}_2'$ there are $1-(\alpha\times g_c(y_1')-(1-\alpha))$ additional number of failed physical nodes. We assume that the effect of removing $1-(\alpha g_c(y_1')-(1-\alpha))$ fraction of nodes in $\mathcal{P}_1$ has the same effect as removing the same amount of nodes from $\mathcal{P}_1'$. As a result, the total number of nodes that needs to be removed is $(1-p_p)+p_p(1-(\alpha \times g_c(y_1') - (1-\alpha)))$, such that the number of residual physical nodes is

$$E|\mathcal{C}_2'| = p_p x_2' \times \tilde{g}_p(x_2')|\mathcal{C}| = x_2'|\mathcal{C}|, \tag{5.16}$$

Accordingly, the total number of functional physical nodes can be evaluated as

$$E|\mathcal{C}_2| = y_2' \times \tilde{g}_p(y_2')|\mathcal{C}| = y_2|\mathcal{C}|. \tag{5.17}$$

- **Step 4 at time $t = 2$:** With respect to the number of functional nodes at Step 3, the expected number of cyber nodes that has social-to-physical dependence to $\mathcal{P}_2$ is given as $E|\mathcal{C}'_2| = \sum_{i=1}^{|P_1|} kP(k) \times E|\mathcal{P}_2| \times \frac{E|\mathcal{C}_1|}{|\mathcal{C}|} = \frac{E|\mathcal{P}_2|}{E|\mathcal{P}_1|} E|\mathcal{C}_1| = \frac{x'_2 \tilde{g}_p(x'_2)}{x_1} E|\mathcal{C}_1|$, which means that from $\mathcal{C}_1$ to $\mathcal{C}'_1$ there are $1 - \frac{x'_2 \tilde{g}_p(x'_2)}{x_1}$ node failures. Again, we consider the effect of removing $1 - \frac{x'_2 \tilde{g}_p(x'_2)}{x_1}$ fraction of nodes in $\mathcal{C}_1$ has the same effect as removing the same amount of nodes from $\mathcal{C}'_1$. As a result, the total number of failed nodes becomes $(1 - y'_1) + y'_1(1 - \frac{x'_2 \tilde{g}_p(x'_2)}{x_1}) = 1 - x'_2 \tilde{g}_p(x'_2)$ and the number of residual cyber nodes becomes

$$E|\mathcal{C}'_2| = p_p x'_2 \times \tilde{g}_p(x'_2)|\mathcal{C}| = x'_2|\mathcal{C}|. \tag{5.18}$$

The total number of functional cyber nodes can then be calculated as

$$E|\mathcal{C}_2| = y'_2 \times \tilde{g}_p(y'_2)|\mathcal{C}| = y_2|\mathcal{C}|. \tag{5.19}$$

By continuing the steps above, we can collect the number of remaining and functional nodes at each time step. Finally, by normalizing $E|\mathcal{P}'_t|$ and $E|\mathcal{P}_t|$ to $|\mathcal{P}|$ as well as $E|\mathcal{C}'_t|$ and $E|\mathcal{C}_t|$ to $|\mathcal{C}|$, we obtain (5.10)-(5.9). □

The expected physical utility ratio $\bar{x}_t$ in Lemma 13 has a decreasing value over time, and its supremum is equal to the node yield Y_n. In other words, the steady-state value of $\bar{x}_t$ amounts to Y_n. As a result, Y_n can be solved by finding the mean fraction of physical residues in (5.10) that satisfies $\bar{x}'_t = \bar{x}'_{t-1}$, and then plugging the result into (5.8).

From Original Physical Graph Into Auxiliary Graph

Lemma 13 provides a way for analyzing the node yield as a function of $\tilde{g}_p$, which according to (5.5)-(5.7) requires the probability generating function (PGF) $\tilde{G}_{p,0}(x)$ of the auxiliary graph $\tilde{\mathcal{G}}_p$. The PGF, however, is unknown and we propose the following mapping-based approach to find it. First of all, we use the degree distribution of the original physical graph $\mathcal{G}_p$ to find its PGF after `Type-1` faults. In order to use Lemma 13, this must be equal to the PGF of the auxiliary graph after random node attack (`Type-0`) fault. Then, we apply an inverse mapping to find the PGF of the auxiliary graph prior to `Type-0` fault. Such a mapping-based approach is detailed (in reverse order) as follows.

In cyber-physical networks, the initial faults at a $DR = 1 - \phi$ fraction of physical edges may cause some nodes to be disconnected. To this end, a node $b_i \in \mathcal{P}$ with a physical degree $d_p(b_i)$ is disconnected when all of its edges are removed, which happens with probability $(1 - \phi)^{d_p(b_i)}$. By applying an expectation with respect to the probability and taking its complement, the initial

fraction of residual nodes is given as

$$p_p = 1 - \sum\nolimits_{k=0}^{\infty} P_p(k)(1-\phi)^k. \tag{5.20}$$

On the other hand, because edge faults occur randomly with probability $DR = 1 - \phi$, the node degree distribution right after `Type-1` the initial fault becomes

$$P_{\phi,p}(k) = \sum\nolimits_{j=k}^{\infty} P_p(j) DR^{j-k}(1-DR)^k. \tag{5.21}$$

This is equal to the degree distribution of auxiliary graph $\tilde{\mathcal{G}}_p$ after `Type-0` (random node) faults. Accordingly, the probability generating function of $\tilde{\mathcal{G}}_p$ right after the initial fault is given as $G_{\phi,p}(x) = \sum_{k=0}^{\infty} P_{\phi,p}(k)x^k$. It is known that if the PGF of the auxiliary graph before fault is $\tilde{G}_{p,0}(x)$, then the PGF after `Type-0` fault is given as [84]

$$G_{\phi,p}(x) = \tilde{G}_{p,0}(1 - p_p + p_p x). \tag{5.22}$$

After performing an inversion to (5.22), we finally obtain

$$\tilde{G}_{p,0}(x) = G_{\phi,p}\left(\frac{x}{p_p} + \left(1 - \frac{1}{p_p}\right)\right). \tag{5.23}$$

Equations (5.20)-(5.23) describe the mapping of a `Type-1` initial faults in the original graph to a `Type-0` (random node) faults in the auxiliary graph. By plugging (5.20)-(5.23) into Lemma 13 and finding the steady-state solution of (5.8)-(5.10) as $t \to \infty$, we can obtain our main metric-of-interest, i.e., the node yield Y_n. Together, Lemma 13 and eqs. (5.20)-(5.23) provide a framework for analyzing the node yield as a function of the PGFs of the physical and cyber graphs, $(\mathcal{G}_p, \mathcal{G}_c)$.

One example of cyber-physical graphs (networks) is the device-to-device (D2D)-based mobile social network (MSN), which enables proximate smart phone users to form ad-hoc networks for exchanging multimedia contents [111]. In D2D-based MSN, the ad-hoc network acts as $\mathcal{G}_p$, the MSN-based social connectivity is the $\mathcal{G}_c$, while the wireless devices and MSN users depend on each other's availability to function. When D2D-based MSN is deployed in a conference, $(\mathcal{G}_p, \mathcal{G}_c)$ can be modeled as Erdös-Rényi graphs with Poisson-distributed node degrees [111]. Under a Poisson distribution assumption, the node yield satisfies the following closed-form expression.

Theorem 10. *Let the node degrees of graphs $\mathcal{G}_p$ and $\mathcal{G}_c$ be Poisson-distributed with mean degrees*

$\bar{k}_p$ *and* $\bar{k}_c$*, respectively. Under* `Type-1` *initial faults, the node yield* Y_n *satisfies*

$$\begin{aligned} Y_n = & \left(1 - e^{-\bar{k}_p \phi}\right) \left(1 - \alpha \exp\left\{-\bar{k}_c Y_n\right\}\right) \\ & \times \left(1 - \exp\left\{-\frac{\bar{k}_p \phi}{1 - \exp(-\bar{k}_p \phi)} Y_n\right\}\right). \end{aligned} \tag{5.24}$$

Proof. The physical graph $\mathcal{G}_p$ has a Poisson degree distribution of $P_p(k) = \frac{e^{-\bar{k}_p}}{k!} \bar{k}_p^k$. After plugging $P_p(k)$ into (5.21), we obtain the degree distribution after initial faults of

$$P_{\phi,p}(k) = \frac{e^{-\bar{k}_p}}{k!} \left(\frac{1 - DR}{DR}\right)^k \sum_{j=k}^{\infty} \frac{1}{(j-k)!} (\bar{k}_p DR)^j \tag{5.25}$$

$$= \frac{e^{-\bar{k}_p \phi}}{k!} (\bar{k}_p \phi)^k. \tag{5.26}$$

Consequently, the probability generating function of node degree after the initial fault is given as

$$G_{\phi,p}(x) = e^{\bar{k}_p \phi (x-1)}. \tag{5.27}$$

According to (5.20), the fraction of non-disconnected nodes is $p_p = 1 - e^{-\bar{k}_p \phi}$. After combining p_p, (5.27), and (5.23), the PGF of the auxiliary graph before the initial faults becomes

$$\tilde{g}_p(x) = 1 - \tilde{G}_{p,0}(1 - x(1 - f_p)) = 1 - f_p, \tag{5.28}$$

since $\tilde{G}_{p,1}(x) = \frac{\tilde{G}'_{p,0}(x)}{\tilde{G}'_{p,0}(1)}$ and $f_A = \tilde{G}_{p,1}(1 - x(1 - f_p)) = \exp(\frac{-\bar{k}_p \phi}{1 - \exp(-\bar{k}_p \phi)} x(1 - f_p))$ by definition.

The cyber graph $\mathcal{G}_c$, on the other hand, has a degree distribution of $P_c(k) = \frac{\bar{k}_c^k}{k!} e^{-\bar{k}_c}$ and a PGF of $G_{c,0}(x) = e^{\bar{k}_c (x-1)}$. Hence, we obtain

$$g_c(y) = 1 - G_{c,0}(1 - y(1 - f_c)) = 1 - f_c, \tag{5.29}$$

with f_c satisfying $f_c = G_{c,1}(1 - y(1 - f_c)) = e^{-\bar{k}_c y (1 - f_c)}$.

At steady state, denote $\bar{x}' := \bar{x}'_t = \bar{x}'_{t-1}$ and $\bar{y}' := \bar{y}'_t = \bar{y}'_{t-1}$ to indicate that the expected utility ratios remain constant. Moreover, we plug (5.28)-(5.29) into Lemma 13 to re-state Y_n as

$$\begin{aligned} Y_n &= \bar{x}' \tilde{g}_p(\bar{x}') = p_p(1 - \alpha(1 - g_c(\bar{y}')) \times \tilde{g}_p(\bar{x}') \\ &= p_p(1 - \alpha[\exp\{-\bar{k}_c \bar{y}'(1 - f_c)\}]) \\ &\quad \times \left(1 - \exp\left\{-\frac{\bar{k}_p \phi}{1 - \exp\{-\bar{k}_p \phi\}} x'(1 - f_p)\right\}\right). \end{aligned} \tag{5.30}$$

Finally, the main result in (5.24) can be obtained by approximating $Y_n \approx p_p(1-f_p)(1-f_c)$ when α is large and plugging $p_p = 1 - e^{-\bar{k}_p \phi}$ into (5.30). □

In (5.24), node yield Y_n is stated as a function of Y_n itself (see the right-hand side), and we refer such a function to be self-consistent. A self-consistent equation can be solved graphically [85, Fig. 2] or via *bisection* method [36, Alg. 4.1].

5.3.2 Node Yield Under Type-2 Initial Faults

After analyzing the node yield under `Type-1` fault, in this section we focus on analyzing the node yield Y_n corresponding to networks with `Type-2` fault, that is, when initial faults occur at both the physical edges and cyber nodes. Specifically, we find the equivalent cyber-physical networks $(\tilde{\mathcal{G}}_p, \mathcal{G}_c)$ such that the initial faults at $(1-p_p)$ and $(1-p_c)$ fractions of *nodes* in $\tilde{\mathcal{G}}_p$ and $\mathcal{G}_c$, respectively, have the same effect as

1. disconnecting $(1-\phi)$ fraction of nodes from the original physical graph $\mathcal{G}_p$, and
2. removing $DR_n = \kappa(1-\phi)$ fraction of nodes form the cyber graph $\mathcal{G}_c$.

Our result is presented as follows.

Corollary 11. *Under `Type-2` fault, the expected utility ratios are given in (5.8)-(5.9), while the mean fraction of residues satisfy*

$$\begin{cases} \bar{x}'_t &= p_p p_c \times \left[1 - \alpha\left(1 - g_c(\bar{y}'_{t-1})\right)\right], \\ \bar{y}'_t &= p_p p_c \times \tilde{g}_p(\bar{x}'_t). \end{cases} \tag{5.31}$$

Proof. We start by considering the initial faults in the equivalent networks. After removing $(1-p_p)$ and $(1-p_c)$ fractions of nodes from $\tilde{\mathcal{G}}_p$ and $\mathcal{G}_c$, the respective graphs have p_p and p_c fractions of residual nodes. The fraction of nodes that are residues in *both* $\tilde{\mathcal{G}}_p$ and $\mathcal{G}_c$ becomes equal to $p_p \times p_c$. As a result, discarding $(1-p_p)$ and $(1-p_c)$ fractions of nodes *separately* is equivalent to removing $(1 - p_p \times p_c)$ fraction of nodes from *either* graph $\tilde{\mathcal{G}}_p$ or graph $\mathcal{G}_c$ only. To be consistent with Section 5.3.1, we assume that the removals occur in $\tilde{\mathcal{G}}_p$.

To proceed, we focus on finding the equivalent networks $(\tilde{\mathcal{G}}_p, \mathcal{G}_c)$ in relation to the initial faults. Since we have *transferred* all faults to the physical graph $\tilde{\mathcal{G}}_p$, then the problem is equivalent to finding the PGF of $\tilde{\mathcal{G}}_p$ as discussed in the proof of Lemma 13. Thus, we can re-use the results in (5.8)-(5.10), but with the fraction of initial residues p_p substituted with $p_p \times p_c$, to obtain Corollary 11. □

With Corollary 11 at hand, we can analyze Y_n under `Type-2` fault. When the node degrees of the cyber-physical graphs are Poisson-distributed, Y_n satisfies the following property.

Lemma 14. *Let graphs $\mathcal{G}_p$ and $\mathcal{G}_c$ have Poisson-distributed node degrees. Under `Type-2` fault, the cyber-physical networks $(\mathcal{G}_p, \mathcal{G}_c)$ has a node yield satisfying*

$$\begin{aligned} Y_n = &\left(1-\kappa(1-\phi)\right)\left(1-e^{-\bar{k}_p\phi}\right)\left(1-\alpha\exp\left\{-\bar{k}_c Y_n\right\}\right) \\ &\times\left(1-\exp\left\{-\frac{\bar{k}_p\phi}{1-\exp(-\bar{k}_p\phi)}Y_n\right\}\right). \end{aligned} \tag{5.32}$$

Proof. From the analysis for `Case 1` above, the node degree distribution's PGF of the auxiliary network before fault is given in (5.28), with $\tilde{g}_p(x)$ stated in (5.28). The PGF of the cyber network, on the other hand, is given as $G_{c,0}(x) = e^{\bar{k}_c(x-1)}$ with $g_c(x)$ in (5.29). We can plug $\tilde{g}_p$ and g_c into Corollary 11, focus on the steady-state condition by assigning $\bar{x}' = \bar{x}'_t = \bar{x}'_{t-1}$, and then employ eq. (5.8) to obtain a self-consistent equation of node yield as follows.

$$Y_n = p_p p_c \times (1-\alpha[1-g_c(\bar{y}')])g_p(\bar{x}'). \tag{5.33}$$

After re-applying the final step in the proof of Theorem 10 to (5.33) and substituting $p_c = 1 - DR_n = 1-\kappa(1-\phi)$ into the result, we finally obtain eq. (5.32). □

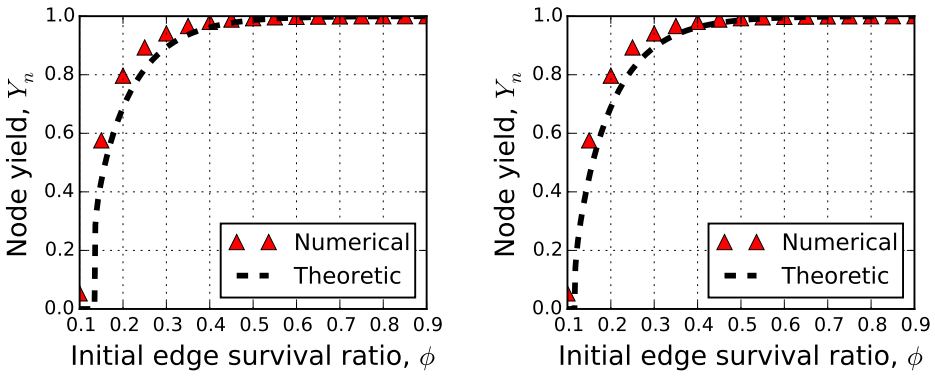
Remark 18. *Compared to (5.24), the node yield of `Type-2` fault in (5.32) has an additional factor of $1-\kappa(1-\phi)$. Since κ and ϕ are non-negatives, the additional factor is smaller than one, and `Type-2` fault has a node yield that is lower than that of `Type-1` fault in (5.24).*

5.3.3 Numerical Evaluation

After analyzing the node yield, we validate our results and show how they can provide guidelines for designing cyber-physical networks that are robust against initial faults.

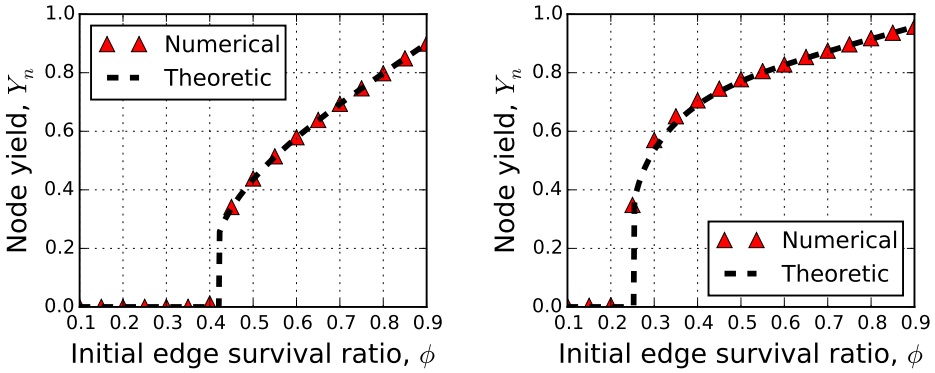
Validation of Analysis

First, we validate our mathematical analysis in Theorem 10. To do this, we employ cyber and physical graphs modeled as Erdös-Rényi (ER) graphs with Binomial degree distributions [58]. It is known that Binomial degree distribution converges to a Poisson for large number of vertices [43]. Unless specified otherwise, we generate ER networks with $n_p = n_s = 5000$ nodes, set the mean physical and cyber degrees to $\bar{k}_p = \bar{k}_c = 10$, and employ the adoption rate of $\alpha = 1$. Each of the presented numerical result is executed in `Python` over 5×10^3 network realizations.



(a) Full adoption ($\alpha = 1$). (b) Partial adoption ($\alpha = 0.2$).

Figure 5.5: Node yield of cyber-physical networks under `Type-1` fault.

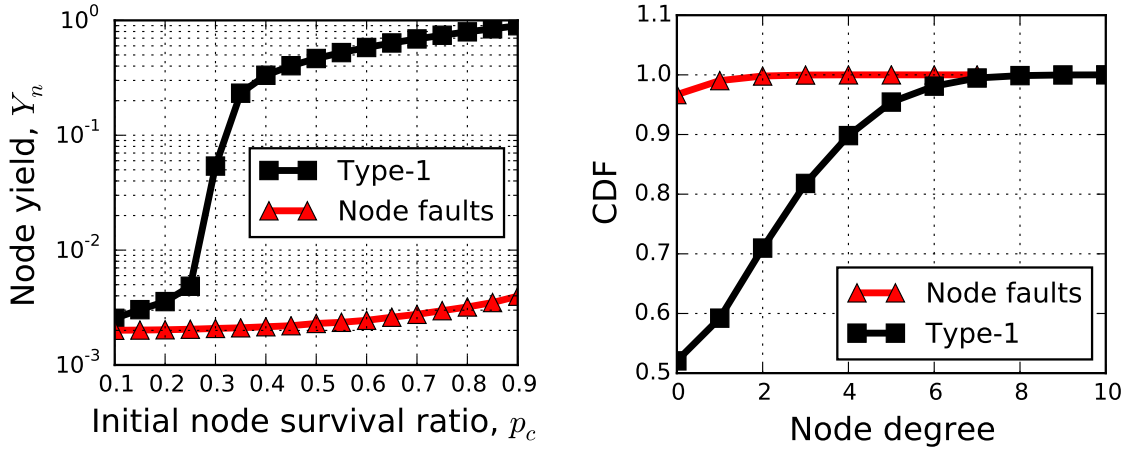


(a) $\kappa = 1$. (b) $\kappa = 0.4$.

Figure 5.6: Node yield of cyber-physical networks under `Type-2` fault.

We plot in Fig. 5.5a the node yield Y_n versus edge survival ratio ϕ under `Type-1` initial fault. The dotted black line represents numerical result, while the red triangles is the solution to the analytical result in (5.24), which is obtained by using *bisection method* [36, Alg. 4.1]. We observe that the former result agree with the latter, indicating that Theorem 10 is valid. The validity is preserved for general α, as indicated by the matching plots in Fig. 5.5b even when α is reduced to 0.2. Moreover, the highly similar plots in Figs. 5.5a and 5.5b reveals that *regardless of the adoption phase, α, node yield under `Type-1` fault will be identical.*

Next, we evaluate Lemma 14 by considering Y_n under `Type-2` fault. We employ two different node-to-line ratios in Figs. 5.6a and 5.6b: $\kappa = 1$ and $\kappa = 0.4$. We observe an agreement between the analytical and numerical solutions, which validates Lemma 14. Moreover, the figures reveal that as κ decreases, the result for `Type-2` fault converges to the node yield of `Type-1` fault in Fig. 5.5a. The convergence is because the latter is a special case of the former with $\kappa = 0$.



(a) `Type-1` versus node fault [38].

(b) Physical node degree after fault.

Figure 5.7: Node yield under faulty edges (`Type-1`) versus cyber node fault.

Practical guideline

Fig. 5.6a indicate that `Type-2` fault, which consists of initial fault at physical edges and cyber nodes, triggers a Y_n that is smaller than that in Fig. 5.5a. When Y_n is smaller, more nodes will eventually fail, which means the networks are more vulnerable. We ask, which fault is the network more vulnerable to: edge or node? To answer the question, we examine cyber-physical networks with initial fault at a $(1-\phi)$ fraction of cyber nodes only [38]. By comparing such a "node fault" scenario to `Type-1` (physical edge) fault in Fig. 5.7a, we observe that Y_n is smaller and the network is more vulnerable against node fault. A closer look at the physical graph's degree distribution after the initial fault (see Fig. 5.7b) reveals that the node fault [38] causes a lower overall degree, making the networks easier to disintegrate. Thus, *it is more important to guard the cyber nodes against initial fault.*

5.4 Resilience of Cyber-Physical Networks

The node yield analysis in Section 5.3 provides a method for quantifying the portion of functional nodes after a cascade. Before evaluating the portion of functioning nodes, however, it is necessary to know *whether* any portion of the cyber-physical networks is still functional, and the required condition for this to happen. In this section, we adopt the notion of *network resilience* to analyze the minimum amount of initial faults that will lead to a non-zero amount of functional nodes.

5.4.1 Preliminaries on Network Resilience

Resilience is a measure of how a network is able to maintain normal operation under faults and attacks. From a graph theoretical perspective, resilience of a network is quantified using *connectivity* [131], that is, the minimum number of edges that has to be removed before the network becomes disconnected. In this paper, we expand the existing notion of connectivity-based resilience for single network to cyber-physical networks with interdependence. Specifically, we focus on the minimum amount of edge faults that leads to a complete system failure, which is indicated by a node yield of $Y_n = 0$. To maintain consistency with previous sections and account for arbitrarily number of edges, we capture the amount of initial faults using the edge disconnection ratio. As a result, our metric-of-interest can be stated as follows.

Definition 31. *Let $Y_n(DR)$ denote the value of node yield as a function of the edge disconnection ratio DR. Then, the critical disconnection ratio is defined as*

$$DR_{cr} := \min\{DR : Y_n(DR) > 0\}. \tag{5.34}$$

For example, the node yield in Fig. 5.6a for $\kappa = 1$ remains zero up to $\phi = 0.421$, which corresponds to a critical disconnection ratio of $DR_{cr} = 1 - 0.421 = 0.579$. A higher DR_{cr} means more edge removals are needed to totally damage the cyber-physical networks. Thus, DR_{cr} is proportional to the networks' resilience against faults.

5.4.2 Main Results

According to Definition 31, DR_{cr} can be found by searching the point where the node resilience Y_n first transitions from positive to zero. Each point of the search requires solving Y_n using Lemma 13 or Corollary 11. For the special case of cyber-physical networks with Binomial degree distributions, DR_{cr} has the following closed-form expression.

Theorem 11. *Consider cyber-physical networks with Binomial degree distributions and a fully-adapted smart physical nodes (i.e., $\alpha = 1$). Then, the critical disconnection ratio is given as*

$$DR_{cr} \approx \begin{cases} 1 - \frac{1.59362}{k_p}, & \text{under } \texttt{Type-1} \text{ fault,} \\ 1 - \frac{\kappa+1}{\kappa+k_p}, & \text{under } \texttt{Type-2} \text{ fault.} \end{cases} \tag{5.35}$$

Proof. We start by considering `Type-1` initial fault. Under `Type-1` fault, solving DR_{cr} is equivalent to finding the smallest edge survival ratio $\phi_{cr} = 1 - DR_{cr}$ that results in a non-zero Y_n, where Y_n satisfies (5.24). To this end, we apply the following two-step approach.

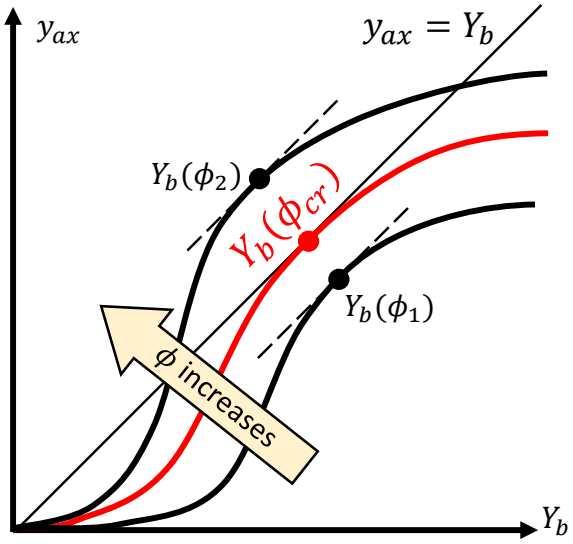


Figure 5.8: A critical condition leading to the critical disconnection ratio DR_{cr}.

(Step 1) By assuming that ϕ is fixed, we find Y_n such that the left-hand side of (5.24) has the same gradient as the right-hand side. In this so called *gradient condition*,

$$\frac{d}{dY_n}Y_n = \frac{d}{dY_n}\left[\left(1-e^{-\bar{k}_p\phi}\right)\left(1-\exp\left\{-\bar{k}_cY_n\right\}\right) \times\left(1-\exp\left\{-\frac{\bar{k}_p\phi}{1-\exp(-\bar{k}_p\phi)}Y_n\right\}\right)\right]. \tag{5.36}$$

To proceed, a first-order Taylor series approximation $e^{-f(Y_n)} \approx 1 - f(Y_n)$ [141] is applied to (5.36) such that the gradient condition becomes

$$1 = \frac{(1-e^{-\bar{k}_p\phi})\bar{k}_p\bar{k}_c\phi}{1-\exp\{-\bar{k}_p\phi\}} \times 2Y_n. \tag{5.37}$$

The solution to (5.37) can be found as

$$Y_n = (2\bar{k}_c\bar{k}_p\phi)^{-1}. \tag{5.38}$$

(Step 2) Next, given Y_n in (5.38) above, we find a critical edge survival ratio ϕ_{cr} to get $DR_{cr} = \phi_{cr}$. To illustrate how this is done, we present Fig. 5.8 in which the curve is plotted for different initial survival ratios, $\phi_1 < \phi_{cr} < \phi_2$. Note that there can be several initial survival ratios ϕ's with their corresponding Y_n's that satisfy the gradient condition. However,

there is only one, ϕ_{cr}, that satisfies the *critical condition*, in which the line and curve intersects non-trivially (i.e. other than at origin) only once. This critical solution can be obtained by plugging (5.38) into (5.24) and re-applying the Taylor series approximation. We obtain $\frac{1}{2\bar{k}_c\bar{k}_p\phi_{cr}} = \bar{k}_p\phi_{cr}\left(\frac{\bar{k}_c}{2\bar{k}_c\bar{k}_p\phi}\right) \times \left(\frac{\bar{k}_p\phi_{cr}}{1-\exp\{-\bar{k}_p\phi_{cr}\}} \times \frac{1}{2\bar{k}_c\bar{k}_p\phi_{cr}}\right)$. After re-arranging and variable eliminations,

$$\bar{k}_p\phi_{cr} = 2(1 - \exp\{-\bar{k}_p\phi_{cr}\}). \tag{5.39}$$

Numerical evaluations [156] show that (5.39) has two solutions: $\bar{k}_p\phi_{cr} = 0$ and $\bar{k}_p\phi_{cr} = 1.59362$. Taking the non-trivial solution (i.e. the latter) and plugging the obtained ϕ_{cr} into $DR_{cr} = 1-\phi_{cr}$ completes the proof for `Type-1` fault.

For cyber-physical networks under `Type-2` fault, we re-employ the Steps 1 and 2 above to the self-consistent equation (5.32). We obtain

$$(1 - \exp\{-\bar{k}_p\phi_{cr}\}) = \bar{k}_p\phi_{cr}(1 - \kappa(1 - \phi_{cr})). \tag{5.40}$$

By applying a second-order Taylor approximation $e^{-f(\phi_{cr})} \approx 1 - f(\phi_{cr}) + \frac{(-f(\phi_{cr}))^2}{2!}$ to the left-hand side of (5.40),

$$2\left(\bar{k}_p\phi_{cr} - \frac{\bar{k}_p^2\phi_{cr}^2}{2}\right) = \bar{k}_p\phi_{cr}(1 - \kappa(1 - \phi_{cr})). \tag{5.41}$$

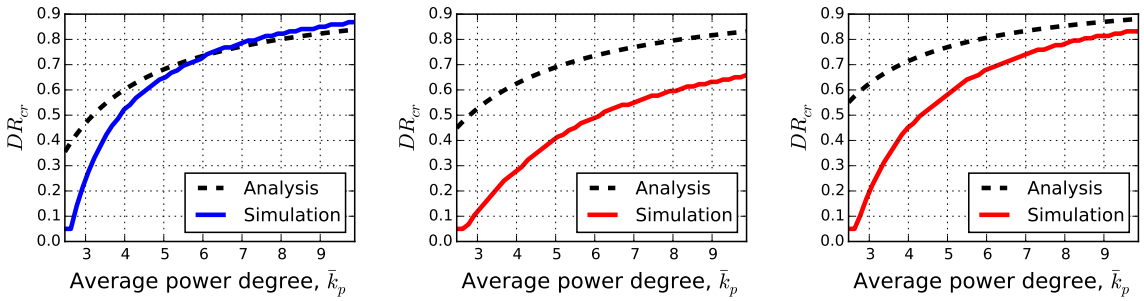
Finally, the second line of (5.35) can be acquired after re-arranging (5.41) and plugging the obtained ϕ_{cr} into $DR_{cr} = 1 - \phi_{cr}$. □

Theorem 11 provides a closed-form expression of DR_{cr} as a function of $\bar{k}_p$ and κ. In the following, we will use numerical simulation to validate Theorem 11 and show how it provides guideline for building resilient cyber-physical networks.

5.4.3 Numerical Validation

To evaluate our analytical DR_{cr} in the first line of (5.35), we consider `Type-1` fault and re-use the simulation parameters in Section 5.3.3. In Fig. 5.9a, the analytical DR_{cr} is compared to the numerical solution that searches DR_{cr} in (5.34) over all its possible values. Notice that there is a slight gap between the analytical and numerical results because eq. (5.37) in the proof of Theorem 11 employs the first-order approximation $e^{-f(Y_n)} \approx 1 - f(Y_n)$, whose error diminishes as Y_n becomes smaller. To this end, Y_n is small near DR_{cr} (see the point in Fig. 5.6a where Y_n abruptly increases from zero) so that the approximation and the resulting expression in the first line of (5.35) are accurate.

Next, we evaluate the analyzed DR_{cr} in the second line of (5.35) by employing `Type-2` fault,



(a) `Type-1` initial fault. (b) `Type-2` with $\kappa = 0.8$. (c) `Type-2` with $\kappa = 0.2$.

Figure 5.9: Illustration and results for critical disconnection ratio analysis.

setting the node-to-line ratio to $\kappa = 0.8$, and plotting the numerical and analytical DR_{cr} versus $\bar{k}_p$ in Fig. 5.9b. We observe a non-trivial gap between the results, which is due to the first and second-order approximations used to obtain (5.35). Although the gap is larger than that for `Type-1` fault in Fig. 5.9a, it becomes smaller as $\bar{k}_p$ increases. When κ is decreased to 0.2, as depicted in Fig. 5.9c, the gap is minimized even further. Despite the analytical-numerical gap, (5.35) provides a useful indication of DR_{cr}'s trend: it increases sublinearly with $\bar{k}_p$ and, under `Type-2` fault, decreases at least linearly versus κ.

Remark 19. *Figs. 5.9a-5.9c indicate that cyber-physical networks become more resilient as the physical graphs becomes more connected. On the other hand, eq. (5.35) re-iterates the importance of guarding the cyber nodes. This can be observed from the increase in DR_{cr} when κ, which is proportional to the number of faulty nodes, becomes smaller.*

5.5 Cascade Length

5.5.1 Preliminaries on Cascade Length

The critical disconnection ratio DR_{cr} and node yield Y_n described *whether* and *what portion of* the cyber-physical networks remain functional, respectively. In addition to these spatial metrics, it is also desirable to know *when* the cascade-of-failures ends, i.e., the temporal metric, which will be useful in the following scenarios.

- **Smart grid:** Researchers have proposed *load shedding* algorithms that let power nodes automatically discard some of their loads, to stop a cascade-of-failures [151]. Under an optimized load shedding strategy that minimizes the amount of shedded demands, it is known that the highest yield is achievable when shedding is applied halfway between the start and the end of a cascade [28]. Knowing when a cascade stops will provide guideline for designing effective load shedding scheme.

- **D2D-based MSN:** Users' quality-of-experience (QoE) can be measured using the likelihood that they do not become disconnected by the cascade-of-failures before their ongoing social networking session is finished [111]. To this end, knowing when a cascade stops not only provides a way to measure QoE, but may also help in the design of QoE-improving schemes.

To quantify the time until a cascade-of-failures ends, we define the following metric.

Definition 32. *Cascade length is defined as*

$$\tau_{CF} := \min\{t \geq 0 : x_t = x_{t+1}\}. \tag{5.42}$$

In other words, the cascade length is τ_{CF} is the amount of time until no more physical node fails. Since the physical utility ratios, x_t and x_{t+1}, are random variables (see Section 5.2.4), then τ_{CF} is also random. As a result, we are interested in the *expected cascade length* $\bar{\tau}_{CF} := E[\tau_{CF}]$.

5.5.2 Main Result

By applying expectation operator to Definition 32, the expected cascade length can be re-stated as the time until there is less than one physical node failures, that is,

$$\bar{x}'_t - \bar{x}'_{t+1} < \frac{1}{n_p}. \tag{5.43}$$

Thus, $\bar{\tau}_{CF}$ can be calculated by i) setting the initial time to $t = 0$, ii) solving (5.10) and advancing the time to $t = t + 1$, iii) repeating steps i)-ii) until the condition (5.43) is satisfied.

Instead of using numerical calculations, it is desirable to have an analytical solution of $\bar{\tau}_{CF}$. The closed-form analytical solution, however, is hard to obtain due to the non-trivial problem of solving $\tilde{g}_p$, g_c, f_p, and f_c that satisfy their self-consistent equations in (5.5)-(5.7). On the other hand, analysis via scaling order, as defined below, is a powerful tool for describing how fast $\bar{\tau}_{CF}$ increases with respect to other network parameters (i.e., mean physical degree $\bar{k}_p$ and edge survival ratio ϕ).

Definition 33 (Scaling order)**.** *Let $\bar{\tau}_{CF}(x)$ be the value of $\bar{\tau}_{CF}$ corresponding to a network parameter x, and $g(x)$ be a convex function of x. Then, $\bar{\tau}_{CF}(x)$ scales with the same order as $g(x)$, denoted as*

$$\bar{\tau}_{CF}(x) = \Theta(g(x)), \tag{5.44}$$

if there exist $c_1, c_2 > 0$ such that $0 \leq c_1 \times g(x) \leq \bar{\tau}_{CF}(x) \leq c_2 \times g(x)$ for all $x \geq x_0$.

In other words, $\bar{\tau}_{CF}(x) = \Theta(g(x))$ if they both have the same rate of increase. With Definition 33 at hand, we can state the following result.

Theorem 12. *The expected cascade length cyber-physical networks with Poisson-distributed degrees and* `Type-1` *initial fault scales as*

$$E[\tau_{cf}] = \begin{cases} \Theta\left(\frac{\log(n_c)}{(1-\alpha)^{1/2}} \exp(\frac{\bar{k}_p \phi_{cr}}{\phi})\right), & \text{for } \phi > \phi_{cr} \\ \Theta\left((1-\alpha)^{-3/2} \exp(\frac{\bar{k}_p \phi}{2\phi_{cr}})\right), & \text{otherwise} \end{cases}. \tag{5.45}$$

Proof. The main difficulty in analyzing the scaling order of $\bar{\tau}_{CF}$ lies in the fact that its closed-form solution is not readily available. Instead, $\bar{\tau}_{CF}$ is defined as the number of iterations until (5.10), which can be re-stated as

$$\bar{x}'_{t+1} = p_p(1 - \alpha[1 - g_c(p_p g_p(\bar{x}'_t))]), \tag{5.46}$$

satisfies (5.43). To find $\bar{\tau}_{CF}$, we omit the time index from $\bar{x}'_t$ and $\bar{x}'_{t+1}$ and graphically plot the left-hand side (LHS) and right-hand side (RHS) of (5.46) in Fig. 5.10. Then, we will count the number of steps until both the LHS and RHS converges (i.e., has a distance less than $\frac{1}{n_p}$ as suggested in (5.43)).

To proceed, let $p_{p,cr} := 1 - \exp(-\bar{k}_p \phi_{cr})$ be the fraction of non-faulted nodes corresponding to ϕ_{cr}, and assume that the actual fraction $p_p := 1 - \exp(-\bar{k}_p \phi)$ is near $p_{p,cr}$. We observe that near the critical point $p_{p,cr}$ (i.e., the point in Fig. 5.10 where the line (LHS) and curves (RHS) starts to intersect non-trivially), the RHS of (5.46) can be approximated into a quadratic function with the form

$$a(p_p)\bar{x}'^2 + b(p_p)\bar{x}' + c(p_p), \tag{5.47}$$

where a, b, and c are some constants. Then, our proof can be divided into two parts as follows.

(Supercritical region with $\phi > \phi_{cr}$) After applying the quadratic approximation in (5.47), the LHS and RHS of (5.46) for the case of $\phi > \phi_{cr}$ is depicted in Fig. 5.11. The expected cascade length $E[\tau_{cf}]$ can be restated as: given an initial fraction of surviving nodes $\bar{x}'_0 = p_p > p_{p,cr}$, we want to find the number of steps (horizontal red lines in Fig. 5.11) until the curve (RHS of (5.46)) and the line (LHS) intersects (see point $\bar{x}'_{\tau_{CF}}$ in Fig. 5.11).

When p_p is near the critical point $p_{p,cr}$, we find from (5.46) that the vertical distance between the curve (RHS) and the line (LHS) is proportional to $(p_p - p_{p,cr})(1-\alpha)$. Next, we subtract the line from the curve to obtain a tilted curve (see the green plot in the inset of Fig. 5.11), which indicates that the curve-to-line horizontal distance scales as $f \sim \sqrt{(1-\alpha)(p_p - p_{p,cr})}$ and the derivative of the tilted plot is $f' \sim \mu\sqrt{(1-\alpha)(p_p - p_{p,cr})}$, where $\mu > 0$ is a constant. According to Fig. 5.11, one step of (5.46) from time t to $t+1$ consists of a horizontal arrow and a vertical

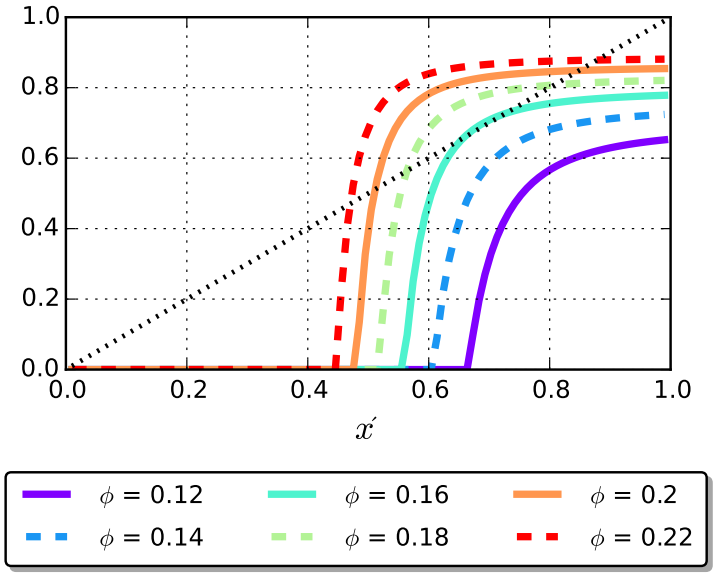


Figure 5.10: Plots of LHS and RHS of (5.46) for various edge survival ratios, ϕ's. Diagonal line represents the LHS, while curves represent the RHS.

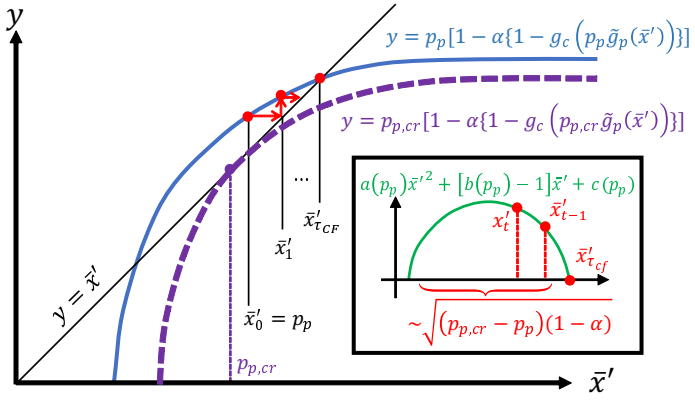


Figure 5.11: Number of steps until cascade for $\phi > \phi_{cr}$.

arrow. We are interested in $\bar{\tau}_{CF}$ that is equal to the number of horizontal arrows. The a horizontal arrow is equal to the number of node failures at each step and has a length proportional to $\exp(\mu\bar{\tau}_{CF}\sqrt{(1-\alpha)(p_p - p_{p,cr})})$. Moreover, the stopping condition is when exactly one node fails, that is,

$$\exp(\mu \times \bar{\tau}_{CF}\sqrt{(1-\alpha)(p_p - p_{p,cr})}) \sim \frac{1}{n_p}. \tag{5.48}$$

After employing $\phi = -\frac{1}{k_p}\ln(1-p_p)$ and Definition 33 into (5.48), the expected cascade length can be shown to scale as the first line of (5.45).

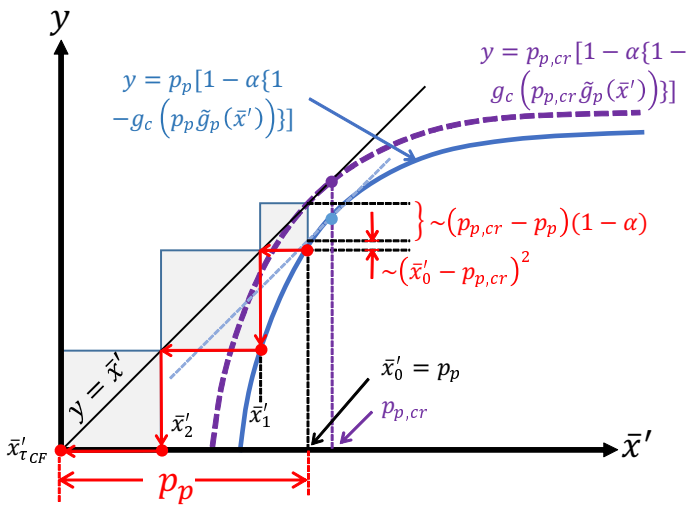


Figure 5.12: Number of steps until cascade for $\phi < \phi_{cr}$.

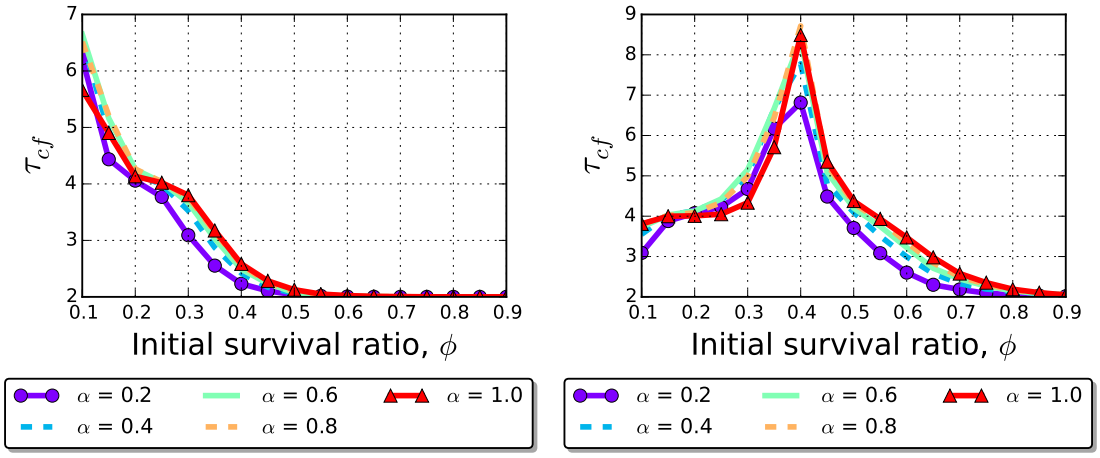
(Subcritical region with $\phi < \phi_{cr}$) According to Fig. 5.12, the shortest vertical distance between the curve (RHS of (5.46)) and the line (LHS) is proportional to $(\bar{x}'_0 - p_{p,cr})^2 + (1-\alpha)(p_{p,cr} - p_p)$. The shortest horizontal distance is also proportional to the same factor since the diagonal line (LHS) has unity gradient. For a horizontal distance of $d\bar{x}'_0$, the number of steps to be taken is proportional to $d\bar{x}'_0/[(\bar{x}'_0 - p_{p,cr})^2 + (1-\alpha)(p_{p,cr} - p_p)]$. For the cascade to stop, the iteration must traverse the horizontal path from $\bar{x}' = p_p$ to $\bar{x}' = 0$, such that the number of required steps is proportional to [74, Sec. 2.103]

$$\begin{aligned} E[\tau_{cf}] &\sim \int_{\bar{x}'=0}^{p} \frac{d\bar{x}'_0}{(\bar{x}'_0 - p_{p,cr})^2 + (1-\alpha)(p_{p,cr} - p_p)} \\ &= \pi/\sqrt{(1-\alpha)^3(p_{p,cr} - p_p)}. \end{aligned} \tag{5.49}$$

Finally, by substituting $\phi = -\frac{1}{k_p}\ln(1-p_p)$ into (5.49), we can get the second line of (5.45). □

In Theorem 12, we presented the scaling law of cascade length under a special assumption of cyber-physical networks with Poisson degree distributions and `Type-1` fault. In the following, we show that Theorem 12 also holds under `Type-2` fault and for networks with power law degree distributions.

5.5.3 Numerical Evaluation



(a) `Type-1` fault. (b) `Type-2` fault.

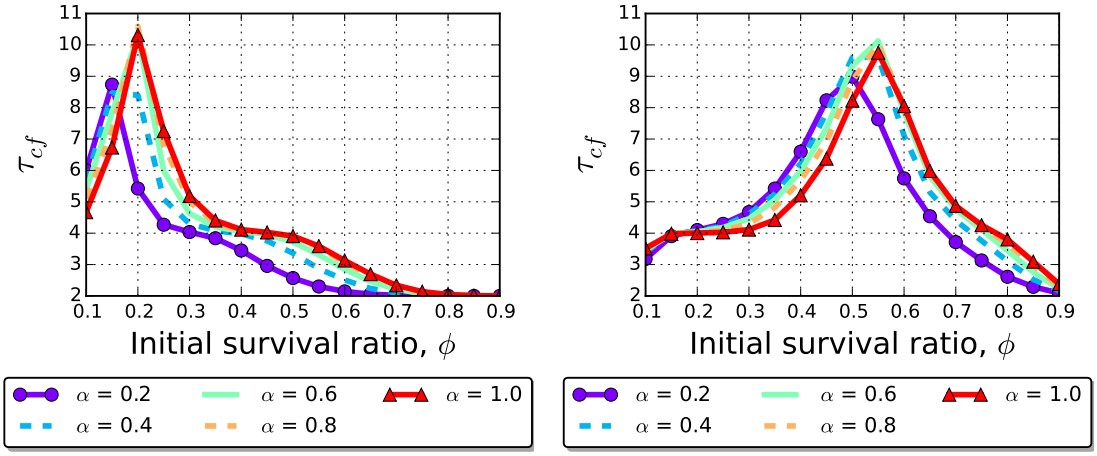
Figure 5.13: Mean cascade length $\bar{\tau}_{CF}$ versus initial survival ratios ϕ for Poisson-distributed cyber-physical networks.

Theorem 12 is evaluated by re-employing the simulation parameters in Section 5.3.3 under `Type-1` fault. In this case, the critical survival ratio ϕ_{cr} (i.e., the smallest ϕ so that the node yield Y_n in Fig. 5.5a first becomes non-zero) is very small. Thus, according to Theorem 12, $\bar{\tau}_{CF}$ should completely scale according to the second line of (5.45). From the resulting cascade length versus edge survival ratio ϕ plot in Fig. 5.13a, we can observe that $\bar{\tau}_{CF}$ decrease according to a scaling order of $\exp(\frac{1}{\phi})$, thus validating the second part of Theorem 12. Note that similar observation also holds when α is increased from 0 to 1.

We proceed by re-applying the cyber-physical networks in Fig. 5.13a but with `Type-2` initial fault and $\kappa = 1$. In this case, the critical survival ratio is equal to $\phi_{cr} = 0.421$ (see Fig. 5.6a). From the $\bar{\tau}_{CF}$ plot in Fig. 5.13b, we observe that up to ϕ_{cr} the cascade length increases according to a factor of $\exp(k \times \phi)$, where k is a constant, thus validating the result in the first row of

(5.45). Otherwise (i.e., when $\phi > \phi_{cr}$), $\bar{\tau}_{CF}$ decreases according to $\exp(\frac{1}{\phi})$, which is similar to that in Fig. 5.13a. The aforementioned scaling laws indicate that Theorem 12 holds not only under `Type-1`, but also `Type-2` initial fault. As α increases, similar results still hold.

Remark 20. *According to Figs. 5.13a and 5.13b, the largest $\bar{\tau}_{CF}$ occurs at the critical point $\phi = 1 - DR_{cr}$. However, the node yield Y_n decreases dramatically when ϕ becomes smaller around the critical point (see Figs. 5.6a and 5.6b). As a result, there is a trade-off near ϕ_{cr}: there is ample time to mitigate the cascade (i.e., $\bar{\tau}_{CF}$ is high), but the spatial impact is much more severe (i.e., Y_n is low).*



(a) `Type-1` fault.

(b) `Type-2` fault.

Figure 5.14: Mean cascade length $\bar{\tau}_{CF}$ versus initial survival ratios ϕ for power law-distributed cyber-physical networks.

Finally, we employ Barabási-Albert (BA) graphs with power law exponents $\lambda_p = \lambda_c = 3$ to represent power law degree-distributed cyber-physical networks. The results for `Type-1` and `Type-2` faults are depicted in Figs. 5.14a and 5.14b, respectively. Compared to the results for Binomial-distributed graphs in Figs. 5.13a and 5.13b, BA graphs have a larger number of high-connected nodes. However, the BA graphs' average degrees (i.e., $\bar{k}_p = \bar{k}_c = 5.96$) are smaller than $\bar{k}_p = \bar{k}_c = 10$ in Figs. 5.13a and 5.13b such that the networks are less robust and the peak of $\bar{\tau}_{CF}$ (i.e., the critical points) occur later. However, we notice that the same scaling order of $\bar{\tau}_{CF}$ as in Figs. 5.13a and 5.13b can be observed, indicating that Theorem 12 also holds for networks with power law degree distributions.

5.6 Summary

We studied how initial fault at physical edges, possibly combined with initial node break downs, can trigger a cascade-of-failures in cyber-physical systems. To model how the cascade progresses, we map the original edge-induced cascades into a well-known node-based cascade problem, which is further analyzed using percolation theory. By introducing and analyzing a node yield metric, which quantifies the portion of functioning nodes after cascades, we show that guarding the physical edges against fault is less important than preventing break down on cyber devices. Moreover, we show that the minimum amount of edge disconnections to completely destroy the systems can be maximized by adding physical edges and protecting cyber devices against break downs. Finally, we found that cascade length is an increasing-then-decreasing function of the amount of disconnected edges, and its maximum value is achieved at the critical disconnection ratio. Our findings provide guidelines for designing cyber-physical systems that are robust against initial faults and break downs.

Chapter 6

Conclusion and Future Directions

In this dissertation, we have presented our research regarding the performance and robustness of interdependent cyber-physical networks against threats. In this chapter, we summarize our main results and discuss the possible future extensions.

6.1 Conclusion

This disseration has been focused on analyzing the performance and robustness of interdependent cyber-physical networks. In Chapter 2, we modeled the access delay performance of cloudlet-based IoT applications in hostile environments. In Chapter 3, we presented how to quickly establish a common control channel between IoT nodes with limited channel hopping capabilities. Chapter 4 detailed the impacts of social and communication-based node dropouts to the connectivity of D2D-based MSNs. In Chapter 5, we investigated how initial edge disconnections can trigger a cascade-of-failures in cyber-physical networks. Next, we summarize and highlight our major findings.

In Chapter 2, we provided an in-depth study regarding the delay in D2D-based IoT applications via analytical modeling and numerical simulations. We proposed a generic *packet mobility* model to capture the progression of data packets toward a fixed point-of-access as the destination, and define access delay as the time until the destination is reached. Then, we divide the packet mobility into *supermartingale*, *submartingale*, and *martingale* regimes in which the packets tend to move closer to, away from, and stay at a constant distance to the destination, respectively. We showed that in the supermartingale regime the access delay scales linearly versus the packet's initial distance-to-destination; for the other regimes, access delay increases quadratically (i.e., faster) with respect to the initial distance. Furthermore, we proposed a *packet shedding* algorithm that discards the generated packets that are predicted to exceed a pre-determined delivery deadline, and show that the shedding scheme can reduce the

total transmission power by up to 24.35%.

In Chapter 3, we studied how to quickly establish a common channel between a pair of IoT-based DSA nodes via channel hopping. To take into account the nodes' limited channel switching time and sensing capacity, we introduced novel *spectrum slice graphs* to describe the set of available channels for the next hop. By viewing the nodes' channel hopping processes as random walks over the spectrum slice graphs and optimizing the walks' transition probabilities, we propose R-RCH and S-RCH schemes that are suitable for IoT nodes with and without distinct pre-determined roles, respectively. Based on numerical evaluation using real-world mobile network traces, we showed that R-RCH and S-RCH outperform the existing MMC scheme and achieve at least 80% successful rendezvous in mobile indoor environment.

In Chapter 4, we considered D2D-based MSNs and investigated the impacts of node dropouts to the quality-of-service, from both the network-wide and end-users' perspective. Using real-world activity traces of Facebook users, we showed that highly-connected MSN nodes are less likely to be dropped out due to their high social activity, unlike in the D2D network where high-degree nodes can be dropped out more often due to malware infections. We analyzed a *minimum utility ratio* metric, which describes the smallest (worst-case) portion of functional nodes after a dropout-induced network contraction process, and showed how to improve the minimum utility ratio by selectively accepting or rejecting new users from joining the D2D-based MSN. Finally, we proposed *node resilience*, which quantifies the likelihood that a user will be disconnected before being able to finish its ongoing MSN session, as a novel quality-of-experience metric. We showed via scaling order analysis that the node resilience can be increased by suppressing the number of of initial dropouts and expanding the networks' size.

In Chapter 5, we studied how edge disconnections in interdependent cyber-physical networks can trigger a cascade-of-failures. We proposed a *node yield* metric to quantify the portion of physical nodes that remain functional after a cascade. By comparing the networks under initial edge disconnections to that with initial node breakdown, we found that the node yield is less sensitive to initial edge disconnections. In other words, guarding edge against disconnections is less important than protecting nodes against initial breakdowns. Additionally, we defined and derived the analytical closed-form solution of the *critical disconnection ratio* to measure the fraction of initial disconnections that will result in a network-wide cascade-of-failures. Finally, we showed via analysis that the cascade length, i.e., the time until a cascade stops, is a non-monotonic exponential function of the edge disconnection ratio and achieves a maximum at a critical disconnection ratio.

6.2 Future Directions

The works presented in this dissertation are focused on improving the performance and understanding how threats can impact cyber-physical networks. In terms of performance improvement, our work in Chapter 3 on the design of fast rendezvous schemes for the lower-priority IoT nodes assumes that the spectrum slice graphs are static. In practice, the spectrum graphs consist of channels that are occupied by PUs, and the occupation can vary along with the PUs' activities. Further study is needed to adapt the proposed R-RCH and S-RCH schemes, or even design a completely new algorithm, for the case of time-varying spectrum slice graphs. Additionally, rendezvous on time-varying graph will benefit from experimental evaluation on real-world test-bed that accounts for channel attenuation, traffic load, false positives and negatives in channel sensing, and erroneous rendezvous signal decoding. A successful test-bed implementation will provide a better guideline for deploying rendezvous schemes in the real world.

In terms of cyber-physical networks' reliability, our current works for quantifying the impacts of node dropouts and edge disconnections in Chapters 4 and 5 are mainly based on an analysis. A joint study from the analysis as well as the design perspectives can offer a more comprehensive understanding towards the performance limits of reliable cyber-physical networks. In particular, we have found in the context of D2D-based MSN that achieving excellent quality-of-service from both the network-wide and end-users' perspectives are desirable. However, the two goals can be paradoxical in practice. For instance, in Chapter 4 we showed that node yield, which quantifies the network-wide connectivity of D2D-based MSN, can be increased by preventing new nodes from joining the networks under severe initial dropouts. The rejected nodes who cannot enjoy any networking service will have a low satisfaction, which further degrades the overall quality-of-experience. To this end, it is desirable to assess the trade-off between the network-wide and end-users' quality-of-service, in order to select the appropriate reliability-enhancing schemes for cyber-physical networks. The trade-off serves as an interesting yet vital piece towards the design of highly-reliable cyber-physical networks.

REFERENCES

[1] Android developers. https://developer.android.com/index.html. Accessed: 2017-02-28.

[2] FireChat. http://opengarden.com/firechat/. Accessed: 2016-03-02.

[3] Firechat alerts. http://www.opengarden.com/alerts.html. Accessed: 2016-09-04.

[4] Firechat sees 100k app downloads a day with a huge uptake overseas. https://gigaom.com/2014/03/26/. Accessed: 2015-07-21.

[5] Google Spectrum Database. https://www.google.com/get/spectrumdatabase/. Accessed: 06-28-2017.

[6] How to switch central frequency fastly with labview and USRP? https://forums.ni.com/t5/USRP-Software-Radio/How-to- switch-central-frequency-fastly-with-labview-and-USRP/td-p/2430394. Accessed: 06-21-2017.

[7] Internet of things (iot) connected devices installed base worldwide. https://www.statista.com/statistics/471264/. Accessed: 05-18-2017.

[8] Internet of things on pace to replace mobile phones as most connected device in 2018. https://www.forbes.com/sites/louiscolumbus/. Accessed: 05-05-2017.

[9] Multipeer Connectivity Framework Reference. https://developer.apple.com/library/ios/. Accessed: 11-07-2016.

[10] Unlocking the potential of the internet of things. http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/the-internet-of-things-the-value-of-digitizing-the-physical-world. Accessed: 05-05-2017.

[11] USRP Hardware Driver and USRP Manual: General Application Notes. http://files.ettus.com/manual/page$_g$*eneral.html. Accessed* : 06 − 28 − 2017.

[12] What are the benefits of the networked world for developing countries. http://cyber.law.harvard.edu/readinessguide/benefits.html. Accessed: 2016-09-03.

[13] Wi-Fi Peer-to-Peer. https://developer.android.com/guide/topics/connectivity/wifip2p.html. Accessed: 10-10-2016.

[14] CRAWDAD's mit/reality (v. 2005-07-01). Downloaded from http://crawdad.org/, July 2005.

[15] CRAWDAD's thlab/sigcomm2009 (v. 2012-07-15). Downloaded from http://crawdad.org/, July 2012.

[16] CRAWDAD's unical/socialblueconn (v. 2015-02-08). Downloaded from http://crawdad.org/, February 2015.

[17] Milton Abramowitz and Irene A Stegun. *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. Courier Corp., 1964.

[18] AccuWeather. *Weekly wrap-up: Blizzard of 2016 ranks 4th among worst winter storms of past 100 years*, Jan 2016.

[19] Mohamed Hossam Ahmed. Call admission control in wireless networks: A comprehensive survey. *IEEE communications Surveys and Tutorials*, 7(1-4):50–69, 2005.

[20] Steve Alpern and Shmuel Gal. *The theory of search games and rendezvous*, volume 55. Springer Science & Business Media, 2006.

[21] Mahdi Asadpour et al. Route or carry: Motion-driven packet forwarding in micro aerial vehicle networks. *IEEE Trans. Mobile Computing*, 16(3):843–856, 2017.

[22] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.

[23] Luca Avena, Markus Heydenreich, Frank den Hollander, Evgeny Verbitskiy, and Willem van Zuijlen. Random walks (lecture notes). Technical report, Technical report, Mathematical Institute, Leiden University.

[24] Paramvir Bahl, Ranveer Chandra, and John Dunagan. Ssch: Slotted seeded channel hopping for capacity improvement in ieee 802.11 ad-hoc wireless networks. In *Proc. ACM MobiCom'04*, pages 216–230. ACM, 2004.

[25] Fan Bai and Ahmed Helmy. A survey of mobility models. *Wireless Adhoc Networks. USC, USA*, 206:147, 2004.

[26] Richard Bell. Maximum supported hopping rate measurements using the universal software radio peripheral software defined radio. In *Proceedings of the GNU Radio Conference*, volume 1, 2016.

[27] Fabrício Benevenuto, Tiago Rodrigues, Meeyoung Cha, and Virgílio Almeida. Characterizing user behavior in online social networks. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, pages 49–62. ACM, 2009.

[28] Andrey Bernstein, Daniel Bienstock, David Hay, Meric Uzunoglu, and Gil Zussman. Power grid vulnerability to geographically correlated failuresanalysis and control implications. In *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*, pages 2634–2642. IEEE, 2014.

[29] Christian Bettstetter. On the minimum node degree and connectivity of a wireless multihop network. In *MobiHoc'02*, pages 80–91. ACM, 2002.

[30] Christian Bettstetter, Hannes Hartenstein, and Xavier Pérez-Costa. Stochastic properties of the random waypoint mobility model. *Wireless Netw.*, 10:555–567, Sep 2004.

[31] Kaigui Bian et al. Maximizing rendezvous diversity in rendezvous protocols for decentralized cognitive radio networks. *IEEE Transactions on Mobile Computing*, 12(7):1294–1307, 2013.

[32] Kaigui Bian, Jung-Min Park, and Ruiliang Chen. A quorum-based framework for establishing control channels in dynamic spectrum access networks. In *Proc. MobiCom'09*, pages 25–36. ACM, 2009.

[33] Patrick Billingsley. *Probability and measure*. John Wiley & Sons, 2008.

[34] Stephen Boyd, Persi Diaconis, Pablo Parrilo, and Lin Xiao. Fastest mixing markov chain on graphs with symmetries. *SIAM Journal on Optimization*, 20(2):792–819, 2009.

[35] Stephen Boyd, Persi Diaconis, and Lin Xiao. Fastest mixing Markov chain on a graph. *SIAM review*, 46(4):667–689, 2004.

[36] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge Univ. Press, 2004.

[37] Sarah Buhr. Messaging App Jott Is Blowing Up Among Junior High And High Schoolers. https://techcrunch.com/2015/06/08/page/2/, June 2015. Accessed: 2017-01-16.

[38] Sergey V Buldyrev et al. Catastrophic cascade of failures in interdependent networks. *Nature*, 464(7291):1025–1028, 2010.

[39] Richard L Burden and J Douglas Faires. 2.1 the bisection algorithm. *Numerical Analysis. Prindle, Weber & Schmidt, Boston, MA.*, 676, 1985.

[40] Moira Burke, Cameron Marlow, and Thomas Lento. Feed me: motivating newcomer contribution in social network sites. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 945–954. ACM, 2009.

[41] Han Cai and Do Young Eun. Crossing over the bounded domain: from exponential to power-law inter-meeting time in manet. In *Proc. ACM Mobicom'07*, 2007.

[42] Tracy Camp, Jeff Boleng, and Vanessa Davies. A survey of mobility models for ad hoc network research. *Wireless Commun. Mobile Computing*, 2(5):483–502, 2002.

[43] George Casella and Roger L Berger. *Statistical inference*, volume 2. Duxbury Pacific Grove, CA, 2002.

[44] Augustin Chaintreau et al. Impact of human mobility on opportunistic forwarding algorithms. *IEEE Trans. Mobile Computing*, 6(6):606–620, June 2007.

[45] Cheng-Shang Chang, Wanjiun Liao, and Tsung-Ying Wu. Tight lower bounds for channel hopping schemes in cognitive radio networks. *IEEE/ACM Transactions on Networking*, 24(4):2343–2356, 2016.

[46] Guey-Yun Chang and Jen-Feng Huang. A fast rendezvous channel-hopping algorithm for cognitive radio networks. *IEEE Communications Letters*, 17(7):1475–1478, 2013.

[47] Guey-Yun Chang, Wen-Hung Teng, Hao-Yu Chen, and Jang-Ping Sheu. Novel channel-hopping schemes for cognitive radio networks. *IEEE Transactions on Mobile Computing*, 13(2):407–421, 2014.

[48] Zhiyuan Cheng et al. Exploring millions of footprints in location sharing services. *ICWSM*, 2011:81–88, 2011.

[49] I-Hsun Chuang, Hsiao-Yun Wu, and Yau-Hwang Kuo. A fast blind rendezvous method by alternate hop-and-wait channel hopping in cognitive radio networks. *IEEE Transactions on Mobile Computing*, 13(10):2171–2184, 2014.

[50] Fan Chung and Linyuan Lu. Concentration inequalities and martingale inequalities: A survey. *Internet Math.*, 3:79–127, Mar 2006.

[51] CNN. FireChat in Hong Kong: How an app tapped its way into the protests. http://www.cnn.com/, 2014. Accessed: 2015-07-15.

[52] Thomas H Cormen. *Introduction to algorithms*. MIT press, 2009.

[53] Luiz A DaSilva and Igor Guerreiro. Sequence-based rendezvous for dynamic spectrum access. In *Proceedings of IEEE DySPAN*, pages 1–7, 2008.

[54] Lucas Davi et al. Privilege escalation attacks on android. In *ISC 2010*, pages 346–360. Springer, 2010.

[55] Android Developers. WiFi Peer-to-Peer. https://developer.android.com/guide/topics/connectivity/wifip2 2017. Accessed: 2017-01-10.

[56] Trajanov Dimitar, Filiposka Sonja, Makraduli Jani, and Grnarov Aksenti. Connection resilience to nodes failures in ad hoc networks. In *12th IEEE MELECON*, volume 2, pages 579–582. IEEE, 2004.

[57] Gaogao Dong et al. Percolation of partially interdependent networks under targeted attack. *Physical Review E*, 85(1):016112, 2012.

[58] Paul Erd6s and A Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hungar. Acad. Sci*, 5:17–61, 1960.

[59] P Erdös and A Rényi. On Random Graphs I. *Publ. Math. Debrecen*, 6:290–297, 1959.

[60] Mustafa Ergen and Pravin Varaiya. Throughput analysis and admission control for ieee 802.11 a. *Mobile networks and Applications*, 10(5):705–716, 2005.

[61] Facebook. Company Info. https://newsroom.fb.com/company-info/. Accessed: 04-20-2017.

[62] Xi Fang, Satyajayant Misra, Guoliang Xue, and Dejun Yang. Smart gridthe new and improved power grid: A survey. *Communications Surveys & Tutorials, IEEE*, 14(4):944–980, 2012.

[63] Niroshinie Fernando, Seng W Loke, and Wenny Rahayu. Mobile cloud computing: A survey. *Future Generation Comput. Syst.*, 29(1):84–106, 2013.

[64] Center for The Study of The Presidency and Congress. *Securing The U.S. Electrical Grid: Understanding The Threats To The Most Critical of Critical Infrastructure, While Securing a Changing Grid*, Oct 2014.

[65] LTE-U Forum. Coexistence Study for LTE-U SDL V1.0 (2015-02). *LTE-U Technical Report*, 2015.

[66] Karen A Frenkel. Panning for science. *Science*, 330(6005):748–749, 2010.

[67] Richard H Frenkiel, BR Badrinath, Joan Borras, and Roy D Yates. The infostations challenge: Balancing cost and ubiquity in delivering wireless data. *IEEE Personal Commun.*, 7(2):66–71, 2000.

[68] Deyun Gao, Jianfei Cai, and King Ngi Ngan. Admission control in ieee 802.11e wireless lans. *IEEE network*, 19(4):6–13, 2005.

[69] Mario Gerla, Eun-Kyu Lee, Giovanni Pau, and Uichin Lee. Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. In *IEEE WF-IoT*, pages 241–246. IEEE, 2014.

[70] Minas Gjoka et al. Walking in facebook: A case study of unbiased sampling of OSNs. In *Proc. IEEE INFOCOM*, pages 1–9. IEEE, 2010.

[71] Bar-Giora Goldberg. PLL Synthesizers: A switching speed tutorial. *Microwave Journal*, 44(9):186–186, 2001.

[72] David J Goodman et al. Infostations: A new system model for data and messaging services. In *VTC'97*, volume 2, pages 969–973. IEEE, 1997.

[73] Appu Goundan, Eric Coe, and Cauligi Raghavendra. Efficient broadcasting in delay tolerant networks. In *GLOBECOM'08*, pages 1–5. IEEE, 2008.

[74] IS Gradshteyn and IM Ryzhik. Table of integrals, series, and products, 1965.

[75] Geoffrey Grimmett and David Stirzaker. *Probability and random processes*. Oxford university press, 2001.

[76] Robin Groenevelt, Philippe Nain, and Ger Koole. The message delay in mobile ad hoc networks. *Performance Evaluation*, 62(1):210–228, 2005.

[77] Matthias Grossglauser and David Tse. Mobility increases the capacity of ad-hoc wireless networks. In *INFOCOM'01*, volume 3, pages 1360–1369. IEEE, 2001.

[78] Jayavardhana Gubbi et al. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660, 2013.

[79] Piyush Gupta and Panganmala R Kumar. The capacity of wireless networks. *IEEE Transactions on information theory*, 46(2):388–404, 2000.

[80] Tao Han and Nirwan Ansari. Offloading mobile traffic via green content broker. *IEEE Internet of things journal*, 1(2):161–170, 2014.

[81] Ghaith Hattab and Mohamed Ibnkahla. Multiband spectrum access: Great promises for future cognitive radio networks. *Proceedings of the IEEE*, 102(3):282–306, 2014.

[82] Roger A Hom and Charles R Johnson. Topics in matrix analysis. *Cambridge UP, New York*, 1991.

[83] Chih-fan Hsin and Mingyan Liu. Hitting time analysis for a class of random packet forwarding schemes in ad hoc networks. *Ad Hoc Networks*, 7(3):500–513, 2009.

[84] Xuqing Huang et al. Robustness of interdependent networks under targeted attack. *Phys. Rev. E*, 83(6):065101, 2011.

[85] Zhen Huang et al. Characterization of cascading failures in interdependent cyber-physical systems. *IEEE Trans. Comput.*, 64(8):2158–2168, 2015.

[86] Daniel Iland and Elizabeth M Belding. Emergenet: Robust, rapidly deployable cellular networks. *IEEE Communications Magazine*, 52(12):74–80, 2014.

[87] CISCO Visual Networking Index. Forecast and methodology, 2014–2019. may 27, 2015. *CISCO Systems*, 2015.

[88] Armann Ingolfsson, Susan Budge, and Erhan Erkut. Optimal ambulance location with random delays and travel times. *Health Care Management Science*, 11(3):262–274, 2008.

[89] Business Insider. Hong Kong Protesters Are Communicating Via A Mobile App That Doesn't Actually Use The Internet. http://www.businessinsider.com/firechat-the-app-hong-kong-uses-to-organize-protests-2014-9, 2014. Accessed: 2017-04-20.

[90] Jean Jacod and Philp Protter. *Probability Essentials*. Springer, 2004.

[91] Sushant Jain, Kevin Fall, and Rabin Patra. Routing in a delay tolerant network. In *SIGCOMM '04*, pages 145–158, New York, NY, USA, 2004. ACM.

[92] Chris Jarabek, David Barrera, and John Aycock. Thinav: truly lightweight mobile cloud-based anti-malware. In *ACM ACSAC'12*, pages 209–218. ACM, 2012.

[93] Gunnar Karlsson, Vincent Lenders, and Martin May. Delay-tolerant broadcasting. *IEEE Transactions on Broadcasting*, 53(1):369–381, 2007.

[94] Jon Kleinberg, Mark Sandler, and Aleksandrs Slivkins. Network failure detection and graph connectivity. In *Proc. 5th SODA*, pages 76–85. SIAM, 2004.

[95] Robert K. Knake. A Cyberattack on the U.S. Power Grid. https://www.cfr.org/report/cyberattack-us-power-grid, 2017. Accessed: 2017-07-28.

[96] Paul Kolodzy and Interference Avoidance. Spectrum policy task force. *Federal Commun. Comm., Washington, DC, Rep. ET Docket*, (02-135), 2002.

[97] Yogesh Reddy Kondareddy and Prathima Agrawal. Synchronized mac protocol for multi-hop cognitive radio networks. In *Proc. IEEE ICC'08*, pages 3198–3202. IEEE, 2008.

[98] Karthik Kumar and Yung-Hsiang Lu. Cloud computing for mobile users: Can offloading computation save energy? *Computer*, 43(4):51–56, 2010.

[99] Jason LeBrun, Chen-Nee Chuah, Dipak Ghosal, and Michael Zhang. Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks. In *VTC'05*, volume 4, pages 2289–2293. IEEE, 2005.

[100] David Asher Levin, Yuval Peres, and Elizabeth Lee Wilmer. *Markov chains and mixing times*. AMS, 2009.

[101] Yong Li, Yurong Jiang, Depeng Jin, Li Su, Lieguang Zeng, and Dapeng Wu. Energy-efficient optimal opportunistic forwarding for delay-tolerant networks. *IEEE Transactions on Vehicular Technology*, 59(9):4500–4512, 2010.

[102] Zhuo Lu, Wenye Wang, and Cliff Wang. How can botnets cause storms? understanding the evolution and impact of mobile botnets. In *INFOCOM'14*, pages 1501–1509. IEEE, 2014.

[103] Zhuo Lu, Wenye Wang, and Cliff Wang. Modeling, evaluation and detection of jamming attacks in time-critical wireless applications. *IEEE Transactions on Mobile Computing*, 13(8):1746–1759, 2014.

[104] Zhuo Lu, Wenye Wang, and Cliff Wang. Modeling, evaluation and detection of jamming attacks in time-critical wireless applications. *IEEE Transactions on Mobile Computing*, 13(8):1746–1759, 2014.

[105] Shashidhar Merugu, Mostafa Hamed Ammar, and Ellen W Zegura. Routing in space and time in networks with predictable mobility. 2004.

[106] I Mitliagkas, ND Sidiropoulos, and A Swami. Convex approximation-based joint power and admission control for cognitive underlay networks. In *Proc. IEEE IWCMC'08*, pages 28–32. IEEE, 2008.

[107] Adilson E Motter and Ying-Cheng Lai. Cascade-based attacks on complex networks. *Physical Review E*, 66(6):065102, 2002.

[108] Collin Mulliner and Jean-Pierre Seifert. Rise of the iBots: Owning a telco network. In *MALWARE'10*, pages 71–80. IEEE, 2010.

[109] Dung T Nguyen et al. Detecting critical nodes in interdependent power networks for vulnerability assessment. *IEEE Transactions on Smart Grid*, 4(1):151–159, 2013.

[110] NIST. Cyber-Physical Systems. https://www.nist.gov/el/cyber-physical-systems, 2017. Accessed: 2017-07-28.

[111] S. A. Pambudi, W. Wang, and C. Wang. On the resilience of D2D-based social networking service against random failures. In *GLOBECOM'16*, pages 1–6, Dec 2016.

[112] Sigit Aryo Pambudi and Wenye Wang. Boundary matters: Impact of finite boundary to packet delay performance in mobile data networks. In *Proc. IEEE ICC*, pages 2748–2753. IEEE, 2014.

[113] Nikolaos A Pantazis, Stefanos A Nikolidakis, and Dimitrios D Vergados. Energy-efficient routing protocols in wireless sensor networks: A survey. *IEEE Communications surveys & tutorials*, 15(2):551–591, 2013.

[114] Roni Parshani, Sergey V Buldyrev, and Shlomo Havlin. Interdependent networks: Reducing the coupling strength leads to a change from a first to second order percolation transition. *Physical review letters*, 105(4):048701, 2010.

[115] Milan Patel, B Naughton, C Chan, N Sprecher, S Abeta, A Neal, et al. Mobile-edge computing introductory technical white paper. *White Paper, Mobile-edge Computing (MEC) industry initiative*, 2014.

[116] N Patel. Handset data traffic (2001-2017). Technical report, Strategy Analytics, June 2013.

[117] Yuval Peres and Perla Sousi. Mixing times are hitting times of large sets. *Journal of Theoretical Probability*, 28(2):488–519, 2015.

[118] Charles Perkins, Elizabeth Belding-Royer, and Samir Das. Ad hoc on-demand distance vector (AODV) routing. Technical report, 2003.

[119] A. Picu, Thrasyvoulos Spyropoulos, and T. Hossmann. An analysis of the information spreading delay in heterogeneous mobility DTNs. In *WoWMoM'12*. IEEE, 2012.

[120] Andreea Picu and Thrasyvoulos Spyropoulos. Dtn-meteo: Forecasting the performance of dtn protocols under heterogeneous mobility. *IEEE/ACM Trans. Networking*, 23(2):587–602, 2015.

[121] Anna-Kaisa Pietiläinen et al. Mobiclique: Middleware for mobile social networking. In *WOSN '09*, pages 49–54, New York, NY, USA, 2009. ACM.

[122] Muhannad Quwaider and Yaser Jararweh. Cloudlet-based efficient data collection in wireless body area networks. *Simulation Modelling Practice and Theory*, 50:57–71, 2015.

[123] Ram Ramanathan and Regina Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *INFOCOM 2000*, volume 2, pages 404–413. IEEE, 2000.

[124] Ramachandran Ramjee, Don Towsley, and Ramesh Nagarajan. On optimal call admission control in cellular networks. *Wireless Networks*, 3(1):29–41, 1997.

[125] Vaibhav Rastogi, Yan Chen, and Xuxian Jiang. Droidchameleon: evaluating android anti-malware against transformation attacks. In *ACM SIGSAC'13*, pages 329–334. ACM, 2013.

[126] Injong Rhee, Minsu Shin, Seongik Hong, Kyunghan Lee, Seongjoon Kim, and Song Chong. CRAWDAD dataset ncsu/mobilitymodels (v. 2009-07-23). Downloaded from http://crawdad.org/ncsu/mobilitymodels/20090723, July 2009.

[127] Sushmita Ruj and Arnab Pal. Analyzing cascading failures in smart grids under random and targeted attacks. In *IEEE 28th AINA*, pages 226–233. IEEE, 2014.

[128] Mahadev Satyanarayanan, Paramvir Bahl, Ramón Caceres, and Nigel Davies. The case for VM-based cloudlets in mobile computing. *IEEE Pervasive Computing*, (4):14–23, 2009.

[129] Mahadev Satyanarayanan, Grace Lewis, Edwin Morris, Soumya Simanta, Jeff Boleng, and Kiryong Ha. The role of cloudlets in hostile environments. *IEEE Pervasive Computing*, 12(4):40–49, 2013.

[130] James Scott, Richard Gass, Jon Crowcroft, Pan Hui, Christophe Diot, and Augustin Chaintreau. CRAWDAD dataset cambridge/haggle (v. 2009-05-29). Downloaded from http://crawdad.org/cambridge/haggle/20090529, May 2009.

[131] Arunabha Sen et al. Fault-tolerance in sensor networks. In *Proceedings of IEEE INFOCOM 2006*, 2006.

[132] Ahmed Serageldin, Hani Alturkostani, and Axel Krings. On the reliability of dsrc safety applications: a case of jamming. In *2013 International Conference on Connected Vehicles and Expo (ICCVE)*, pages 501–506. IEEE, 2013.

[133] C-F Shih, Tsung Ying Wu, and Wanjiun Liao. Dh-mac: A dynamic channel hopping mac protocol for cognitive radio networks. In *Proc. IEEE ICC'10*, pages 1–5. IEEE, 2010.

[134] Rostam Shirani et al. The performance of greedy geographic forwarding in unmanned aeronautical ad-hoc networks. In *CNSR'11*, pages 161–166. IEEE, 2011.

[135] Rostam Shirani et al. Quadratic estimation of success probability of greedy geographic forwarding in unmanned aeronautical ad-hoc networks. In *VTC Spring'12*, pages 1–5. IEEE, 2012.

[136] Tara Small and Zygmunt J Haas. The shared wireless infostation model: a new ad hoc networking paradigm (or where there is a whale, there is a way). In *MobiHoc'03*, pages 233–244. ACM, 2003.

[137] Smartgrid.gov. *What is the Smart Grid?*, 2016 (accessed June 30, 2016).

[138] Saleh Soltan, Dorian Mazauric, and Gil Zussman. Cascading failures in power grids: analysis and algorithms. In *Proceedings of the 5th international conference on Future energy systems*, pages 195–206. ACM, 2014.

[139] Thrasyvoulos Spyropoulos, K. Psounis, and C.S. Raghavendra. Single-copy routing in intermittently connected mobile networks. In *SECON'04*, pages 235–244, 2004.

[140] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra. Efficient routing in intermittently connected mobile networks: The multiple-copy case. *IEEE/ACM Trans. Netw.*, 16:77–09, Feb 2008.

[141] Dirk Jan Struik. *A source book in mathematics, 1200-1800*. Princeton University Press, 2014.

[142] Zhanwei Sun and J Nicholas Laneman. Performance metrics, sampling schemes, and detection algorithms for wideband spectrum sensing. *IEEE Transactions on Signal Processing*, 62(19):5107–5118, 2014.

[143] Nick C Theis, Ryan W Thomas, and Luiz A DaSilva. Rendezvous for cognitive radios. *IEEE Transactions on Mobile Computing*, 10(2):216–227, 2011.

[144] Kurt Thomas and David M Nicol. The koobface botnet and the rise of social malware. In *Malicious and Unwanted Software (MALWARE), 2010 5th International Conference on*, pages 63–70. IEEE, 2010.

[145] Amin Vahdat et al. Epidemic routing for partially connected ad hoc networks. Technical report, CS-200006, Duke University, 2000.

[146] Luke Vilnis. Markov Chain Monte Carlo, Mixing, and the Spectral Gap. 2013.

[147] Bimal Viswanath et al. On the evolution of user interaction in facebook. In *WOSN'09*, pages 37–42. ACM, 2009.

[148] Shaoxuan Wang and S. Dey. Adaptive mobile cloud computing to enable rich mobile multimedia applications. *IEEE Trans. Multimedia*, 15(4):870–883, 2013.

[149] Yapeng Wang, Xu Yang, Yutian Zhao, Yue Liu, and Laurie Cuthbert. Bluetooth positioning using rssi and triangulation methods. In *CCNC'13*, pages 837–842. IEEE, 2013.

[150] Lili Wei, Rose Qingyang Hu, Yi Qian, and Geng Wu. Enable device-to-device communications underlaying cellular networks: challenges and research aspects. *IEEE Communications Magazine*, 52(6):90–96, 2014.

[151] Mingkui Wei and Wenye Wang. Dominoes with communications: On characterizing the progress of cascading failures in smart grid. In *2016 IEEE International Conference on Communications (ICC)*. IEEE, 2016.

[152] Eric W Weisstein. Least squares fitting. 2002.

[153] Douglas H Werner and Zhi Hao Jiang. *Electromagnetics of Body Area Networks: Antennas, Propagation, and RF Systems*. John Wiley & Sons, 2016.

[154] Douglas Brent West et al. *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River, 2001.

[155] Christo Wilson, Bryce Boe, Alessandra Sala, Krishna PN Puttaswamy, and Ben Y Zhao. User interactions in social networks and their implications. In *Proceedings of ACM Eurosys*, pages 205–218. Acm, 2009.

[156] Wolfram. WolframAlpha. http://www.wolframalpha.com/, 2016. Accessed: 2016-04-20.

[157] Anna Wu, Xin Yan, and Xiaolong Luke Zhang. Geo-tagged mobile photo sharing in collaborative emergency management. In *Proceedings ACM VINCI'11*, page 7. ACM, 2011.

[158] Linlin Wu, Saurabh Kumar Garg, and Rajkumar Buyya. Sla-based admission control for a software-as-a-service provider in cloud computing environments. *Journal of Computer and System Sciences*, 78(5):1280–1299, 2012.

[159] Shan-Hung Wu, Ching-Chan Wu, Wing-Kai Hon, and Kang G Shin. Rendezvous for heterogeneous spectrum-agile devices. In *Proceedings of IEEE INFOCOM*, pages 2247–2255. IEEE, 2014.

[160] Hongda Xiao and Edmund M Yeh. Cascading link failure in the power grid: A percolation-based analysis. In *IEEE ICC 2011*, pages 1–6. IEEE, 2011.

[161] Liang Xiao, Caixia Xie, Tianhua Chen, Huaiyu Dai, and H Vincent Poor. A mobile offloading game against smart attacks.

[162] Fei Xing and Wenye Wang. Modeling and analysis of connectivity in mobile ad hoc networks with misbehaving nodes. In *ICC'06*, volume 4, pages 1879–1884. IEEE, 2006.

[163] Fei Xing and Wenye Wang. On the expected connection lifetime and stochastic resilience of wireless multi-hop networks. In *GLOBECOM'07*, pages 1263–1267. IEEE, 2007.

[164] Osman Yağan et al. Optimal allocation of interconnecting links in cyber-physical systems: Interdependence, cascading failures, and robustness. *IEEE Transactions on Parallel and Distributed Systems*, 23(9):1708–1720, 2012.

[165] Bo Yang, Meng Zheng, and Wei Liang. A time-efficient rendezvous algorithm with a full rendezvous degree for heterogeneous cognitive radio networks. In *Proceedings of IEEE INFOCOM*, pages 1–9. IEEE, 2016.

[166] D Yang, J Shin, and C Kim. Deterministic rendezvous scheme in multichannel access networks. *Electronics Letters*, 46(20):1402–1404, 2010.

[167] Jie Yang, Yingying Chen, Wade Trappe, and Jerry Cheng. Detection and localization of multiple spoofing attackers in wireless networks. *IEEE Transactions on Parallel and Distributed systems*, 24(1):44–58, 2013.

[168] Yaling Yang and Robin Kravets. Contention-aware admission control for ad hoc networks. *IEEE Transactions on Mobile Computing*, 4(4):363–377, 2005.

[169] Zhi Yang et al. Uncovering social network sybils in the wild. *ACM TKDD*, 8(1):2, 2014.

[170] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer networks*, 52(12):2292–2330, 2008.

[171] Andrea Zanella, Nicola Bui, Angelo Castellani, Lorenzo Vangelista, and Michele Zorzi. Internet of things for smart cities. *IEEE Internet of Things journal*, 1(1):22–32, 2014.

[172] W. Zhang, Y. Wen, and D. O. Wu. Collaborative task execution in mobile cloud computing under a stochastic wireless channel. *IEEE Transactions on Wireless Communications*, 14(1):81–93, Jan 2015.

[173] Xi Zhang and Hang Su. Cream-mac: Cognitive radio-enabled multi-channel mac protocol over dynamic spectrum access networks. *IEEE Journal of Selected Topics in Signal Processing*, 5(1):110–123, 2011.

[174] Xiaolan Zhang et al. Performance modeling of epidemic routing. *Comput. Netw.*, 51(10):2867–2891, July 2007.

[175] Yangfan Zhou, Michael R Lyu, Jiangchuan Liu, and Hui Wang. Port: a price-oriented reliable transport protocol for wireless sensor networks. In *IEEE ISSRE'05*, pages 10–pp. IEEE, 2005.

[176] Daniel Zwillinger. *Table of integrals, series, and products*. Elsevier, 2014.