ABSTRACT

XU, YI. Understanding the Performance and Resilience of Large-Scale Multi-Hop Wireless Networks. (Under the direction of Professor Wenye Wang.)

Wireless networks are becoming an important supplementary technology to the traditional wired networks. They offer convenient and flexible network access for the users to communicate with each other. However, wireless networks confront many technical challenges that limit their full utilization. Especially in large-scale multi-hop networks, the communication quality received by each user depends highly on the cooperation of other users in the network, which is constrained by many factors such as the heterogeneity of user communication devices, the limited availability of radio bandwidth, the difficulty in user coordinations, the mobility of users, and the failure of user devices.

We intend to understand the performance and resilience of large-scale multi-hop wireless networks in this dissertation, which will help us utilize the wireless networks effectively, efficiently and reliably. We identify four fundamental performance and resilience aspects to investigate, namely, the information propagation speed, the communication capacity, the topological stability, and the failure resilience. The study on the first two perspectives attempts to minimize the delay and maximize the capacity of large wireless networks, while the study on the last two perspectives evaluates and mitigates the impact of user mobility and failure on the network structure.

Specifically, we make the following contributions toward improving the utilization of large-scale wireless networks. First, we have determined the maximum information propagation speed in wireless networks and designed a new routing algorithm to identify the minimum transmission delay path for fastest information delivery. Second, we have demonstrated that the maximum network capacity can be obtained by scheduling user transmissions in localized areas and proposed a practical solution for capacity maximization. Third, we have analyzed the network topological stability with presence of user mobility and developed methods to extend the network topology lifetime. Last, we have characterized the spread of correlated user failures and suggested strategies to prevent failures from wide spreading in large wireless networks. The work in this disseration advances our understanding and enhances our utilization of large-scale multi-hop wireless networks.

Understanding the Performance and Resilience of Large-Scale Multi-Hop Wireless Networks

by
Yi Xu

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

Computer Engineering

Raleigh, North Carolina

2010

APPROVED BY:

Dr. Arne A. Nilsson	Dr. Yannis Viniotis
Dr. Wenye Wang Chair of Advisory Committee	Dr. Khaled Harfoush

DEDICATION

To my parents, wife and son.

BIOGRAPHY

Yi Xu was born in Wuhan, China in 1976. He received the B.E. degree in electrical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 1998 and the M.E. degree in computer engineering from the National University of Singapore, Singapore, in 2002, respectively. Since 2005, he has been working toward the Ph.D. degree in the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, USA. Before joining the Ph.D. program, he worked as a research engineer at the Institute for Infocomm Research, Singapore, from November 2001 to July 2005. His research interests include modeling and analysis of large-scale wireless networks, network resilience to mobility and failure, and network security. He has been a member of IEEE since 2000.

ACKNOWLEDGMENTS

First of all, I would like to express my wholehearted gratitude to my academic advisor, Professor Wenye Wang, for her guidance and support throughout my Ph.D. study. For many times in the progress of my research work, it was her enlightening thoughts and patient advices that helped me overcome the difficulties when I was challenged by the research problems. Especially at times when I felt frustrated, she was always my source of encouragement to reestablish my confidence and recover my momentum to move on. I feel myself so fortunate to have the opportunity to learn from her research passion, vision, widsom, and much more, the attitude toward life. I believe that the knowledges and skills that she has imparted to me are far beyond this dissertation. I owe a lifetime-long appreciation to her for the education that I have received.

I would also like to thank the professors in my advisory committee, Professor Arne A. Nilsson, Professor Yannis Viniotis, and Professor Khaled Harfoush, for their support to my research work. Their valuable and insightful suggestions and comments have helped me broaden my research view and improve my work quality. I am also grateful to all the professors that have taught me in and out of classes at NC State for imparting their knowledges, which allows me to be prepared for my future career.

My special thanks also go to my labmates and friends: Ming Zhao, Fei Xing, Nurcan Tezcan, Avesh Algawal, Shawqi Kharbash, Xinbing Wang, Lei Sun, Entong Shen, Chi Yi, Zhuo Lv, Mohit Khanna, Yujin Li, Haiyang Zheng, and Xiang Lu. It has been my great pleasure to work closely with them. My research discussions with them have always been one of my most important sources of fresh ideas that I have benefited from significantly. The time spent with them will remain a great memory for me.

Finally, I am indebted to my parents for their unbounded care and love throughout my life. They taught me the creed of persistence in pursuing dreams and encouraged me in every stage of my pursuit. I am also very thankful to my dear wife for her understanding and support in all the years of my study. She has always been standing besides me to face every challenge and difficulty that we have encountered. Our son, who is curiously discovering this world as he grows up, is my role model for keeping the inquisitive spirit in my future research work. My last special thanks go to this little one.

TABLE OF CONTENTS

LIST OF TABLES . . . . . **viii**

LIST OF FIGURES . . . . . **ix**

Chapter 1 Introduction **1**
1.1 Motivation 1
1.2 Research Objectives 8
1.2.1 Understanding the Information Propagation Speed 8
1.2.2 Understanding the Communication Capacity 10
1.2.3 Understanding the Topological Stability with Node Mobility 11
1.2.4 Understanding the Network Resilience against Correlated Failures . 12
1.3 Contributions 13
1.4 Organization 13

Chapter 2 The Speed of Information Propagation in Wireless Networks . . **15**
2.1 Motivation and Related Work 15
2.2 Problem Formulation 20
2.2.1 Reference Network Model 21
2.2.2 Information Propagation Speed 22
2.3 The Speed Upper Bound 24
2.3.1 Bound in Broadcast Communications 24
2.3.2 Bound in Unicast Communications 25
2.3.3 Solution of Optimal Transmission Radius 31
2.4 The Speed Upper Bound under Network Connectivity Constraint 33
2.4.1 γ-Feasible Packet Delivery 33
2.4.2 Noise Models 34
2.4.3 γ-Feasible Speed Upper Bound 35
2.4.4 Comparison of the Noise Models 37
2.5 The Gap between Achieved Speed and Upper Bound 39
2.6 The Fastest Packet Transmission Algorithm 48
2.7 Summary 49

Chapter 3 The Capacity of Wireless Networks **51**
3.1 Motivation and Related Work 52
3.2 Problem Formulation 53
3.2.1 Network Models 54
3.2.2 Capacity Formulation 56
3.3 Main Results on Network Capacity 58
3.4 Sufficient Condition for Maximum Capacity 61
3.4.1 Link Rate with Global Scheduling 61

3.4.2 Link Rate with Localized Scheduling . . . 63
3.5 Necessary Condition for Maximum Capacity . . . 72
3.6 Design of Localized Scheduling Scheme . . . 75
3.6.1 Distributed Partition Protocol . . . 75
3.6.2 Localized Scheduling Algorithm . . . 81
3.7 Relax of Time Synchronization . . . 85
3.7.1 Clock Non-Synchronization in Partitioning . . . 85
3.7.2 Clock Non-Synchronization in Scheduling . . . 86
3.8 Summary . . . 86

Chapter 4 The Topological Stability of Hierarchical Wireless Networks . . . 88
4.1 Motivation and Related Work . . . 88
4.2 Models and Assumptions . . . 90
4.2.1 Mobility Model . . . 90
4.2.2 Network Model . . . 90
4.3 Lifetime Analysis of the Hierarchical Architecture . . . 91
4.3.1 Cluster Lifetime . . . 91
4.3.2 Inter-Cluster Link Lifetime . . . 98
4.3.3 Path Lifetime . . . 100
4.3.4 Numerical Results . . . 101
4.4 The Mobility and Energy Aware Clustering Algorithm . . . 103
4.5 The Clustered Network Overlay Routing Protocol . . . 106
4.6 Performance Evaluation . . . 108
4.7 Summary . . . 112

Chapter 5 The Wireless Network Resilience against Correlated Failures . . 113
5.1 Motivation and Related Work . . . 113
5.2 Problem Formulation . . . 115
5.2.1 Network and Failure Models . . . 116
5.2.2 Preliminaries of Percolation . . . 119
5.2.3 Failure Percolation . . . 120
5.3 Spread of One-Time Failures . . . 122
5.3.1 Percolation with Constant Failure Impact Radius . . . 123
5.3.2 Percolation with Random Failure Impact Radius . . . 123
5.3.3 Simulation Results . . . 126
5.4 Spread of Persistent Failures: Constant Failure Correlations . . . 127
5.4.1 Sufficient Condition of Percolation . . . 128
5.4.2 Sufficient Condition of Non-Percolation . . . 132
5.4.3 Simulation Results . . . 134
5.5 Spread of Persistent Failures: General Failure Correlations . . . 134
5.5.1 Sufficient Condition of Percolation . . . 136
5.5.2 Sufficient Condition of Non-Percolation . . . 137
5.5.3 Simulation Results . . . 138
5.6 Applications in Failure Control . . . 139
5.7 Summary . . . 140

Chapter 6 Conclusion and Future Directions 141
6.1 Conclusion 141
6.2 Future Directions 143

Bibliography 145

LIST OF TABLES

Table 2.1 Comparison between $y(\alpha)$ and $e^{\alpha} - 1$ 32

Table 4.1 Mobility Patterns 100

LIST OF FIGURES

Figure 2.1 Information propagation in multihop wireless networks. The packet broadcasting duration is τ for each node. 23

Figure 2.2 The packet relay path in direction φ. 24

Figure 2.3 The packet transportation time via 1-hop and 2-hop relay paths, $B = 100$ KHz, $L = 1024$ bits, $\frac{P}{N} = 10^3$, $\alpha = 2$. 28

Figure 2.4 The node removal and relocation process. (a) Finding the node set $\{v_j \mid d_{v_i v_j} \leq d^*, j = i+1, \cdots, i+k\}$. (b) Removing the nodes $\{v_j, j = i+1, \cdots, i+k-1\}$. (c) Relocating the node v_{i+k} such that $d_{v_i v_{i+k}} = d_{v_{i+k} v_{i+k+1}}$ (if $k > 0$) or the node v_{i+1} such that $d_{v_i v_{i+1}} = d_{v_{i+1} v_{i+2}}$ (if $k = 0$). 30

Figure 2.5 The end-to-end packet transportation time, $B = 100$ KHz, $L = 1024$ bits, $\frac{P}{N} = 10^3$, $\alpha = 2$. When $d_{v_0 v_d}$=2m, R_u=2m; when $d_{v_0 v_d}$=15m, 30m, 60m, R_u=15m; when $d_{v_0 v_d}$=120m, R_u=17.14m. 32

Figure 2.6 The optimal transmission radius R. 33

Figure 2.7 $W_\gamma(\lambda)$ in the constant-interference model, $B = 100$ KHz, $L = 1024$ bits, $\frac{P}{N} = 10^3$, $\lambda_C = 50$. 38

Figure 2.8 $W_\gamma(\lambda)$ in the increasing-interference model, $B = 100$ KHz, $L = 1024$ bits, $\frac{P}{N(1)} = 10^3$, $\lambda_I = 50$. 38

Figure 2.9 Information propagation in multihops in direction φ. 40

Figure 2.10 A comparison of the packet propagation speeds in six randomly chosen directions, in which the normalized speed is defined as the ratio of the minimum speed to the maximum speed in the six directions, $\lambda = 30$. 41

Figure 2.11 Definitions of X_{max} and X_{min} 43

Figure 2.12 The propagation distance Z_m in direction φ. 44

Figure 2.13 The mth-hop propagation distance $\rho_m = Z_m - Z_{m-1}$ in direction φ, $d_{z_{m-1}u_4} = d_{z_{m-1}u_5} = d_{z_{m-1}u_6} = d_{u_1u_2} = d_{u_1u_3} = R_\gamma(\lambda)$, $\angle u_4 z_{m-1} u_6 = \angle u_5 z_{m-1} u_6 = \frac{\pi}{3}$ 46

Figure 2.14 The speed gap in the constant-interference noise model, $\frac{P}{N} = 10^3$ 46

Figure 2.15 The speed gap in the increasing-interference noise model, $\frac{P}{N(1)} = 10^3$ 46

Figure 2.16 The fastest packet transmission algorithm running on node v_i that forwards a packet to destination v_d. The vectors $\mathbf{v_i}$ and $\mathbf{v_d}$ denote the locations of v_i and v_d. The environmental parameters α, B, P and N are constant and known. 49

Figure 3.1 Density determination of the concurrent senders in the physical model. The disk has radius $\frac{1}{2}|X_i - X_j|$. The square has edge length $\frac{1}{2\sqrt{2}}|X_i - X_j|$. 62

Figure 3.2 Determination of the bounds on $r(n)$ in the generalized physical model... 63

Figure 3.3 The relative locations of the co-centric disks and an overlapping partition. The co-centric disks are centered at o_1, denoted as $D(o_1, \kappa_1 r)$ and $D(o_1, \kappa_3 r)$ respectively. The annulus is denoted as $D(o_1, \kappa_3 r) \setminus D(o_1, \kappa_1 r)$. The partition is centered at o_2. For clarity, we have only shown the inner disk of radius ρ, denoted as $D(o_2, \rho)$, but ignored the partition boundary as well as the outer disk of radius $\sigma\rho$. Point a is inside $D(o_1, \kappa_1 r)$. Lines $\overline{ab}$ and $\overline{ac}$ are tangent to $D(o_2, \rho)$. Lines $\overline{ad}$ and $\overline{ae}$ are tangent to $D(o_3, \epsilon)$. $d = |\overline{o_1 o_2}|$. 64

Figure 3.4 Proof of Theorem 3.6 in the protocol interference model. In unicast, we define disks $D(v_j, (1+\Delta)r(n))$, $D(v_j, 2(1+\Delta)r(n))$, $D(o_3, \epsilon)$ and $D(o_4, 4(1+\Delta)r(n))$. In broad/multi-cast, we define disks $D(v_i, (2+\Delta)r(n))$, $D(v_i, 2(1+\Delta)r(n))$, $D(o_3, \epsilon)$ and $D(o_4, 4(1+\Delta)r(n))$. The area of $D(o_3, \epsilon)$ is at least $c_2 r^2(n)$. The polygon denotes a neighbor partition. 67

Figure 3.5 Node locations on grids. 68

Figure 3.6 Aggregate interference from the region bounded between distance a and b. 70

Figure 3.7 Proof of Theorem 3.7 in the protocol interference model. Case (a) proves $\rho(n) = \Omega(r(n))$, in which we define disks $D(v_i, 2\sigma\rho(n))$ and $D(p, 2\sigma\rho(n))$. Case (b) proves $\xi(n) = \Omega(r(n))$, in which we define disk $D(p, \xi(n))$. In both cases, $|\overline{v_i p}| = \frac{\Delta r(n)}{2}$. 73

Figure 3.8 The process of determining the role of node v_i. 77

Figure 3.9 Probability distribution of the partition sizes normalized against $r(n)$. ... 79

Figure 3.10 Upper and lower bounds on the partition sizes normalized against $r(n)$... 80

Figure 3.11 Protocol convergence. 80

Figure 3.12 Protocol overhead. 81

Figure 3.13 A localized scheduling algorithm that runs in each partition and in each time slot t. 81

Figure 3.14 Scaling of partition size, $\rho(n) = 1$. 83

Figure 3.15 Scaling of partition size, $\rho(n) = r(n)$. 83

Figure 3.16 Scaling of scheduling density, $\delta(n) = r^{\frac{3}{2}}(n)$. 84

Figure 3.17 Scaling of scheduling density, $\xi(n) = r^{\frac{1}{2}}(n)$. 84

Figure 4.1 The intervals of pauses and movements. 92

Figure 4.2 The movements inside and beyond the coverage area of another node. 92

Figure 4.3 The Markov transition diagram. 96

Figure 4.4 The connection types and the logical link. 98

Figure 4.5 $E(T_m)$. 102

Figure 4.6 $E(T_h)$. 102

Figure 4.7 $E(T_l)$. 102

Figure 4.8 $E(T_p)$. 102

Figure 4.9 The MEACA clustering algorithm. 104

Figure 4.10 An example connection table. 107

Figure 4.11 The node status determination time (MEACA). 108

Figure 4.12 The ripple effect of Lowest-ID and GDMAC. 109

Figure 4.13 The time distribution of the clusterhead status (MEACA). 109

Figure 4.14 The time distribution of the clusterhead status (Lowest-ID). 110

Figure 4.15 The time distribution of the clusterhead status (GDMAC). 110

Figure 4.16 $E(T_m)$. 111

Figure 4.17 $E(T_h)$. 111

Figure 4.18 $E(T_l)$. 111

Figure 4.19 $E(T_p)$. 111

Figure 4.20 Throughput comparison. 112

Figure 5.1 Failure spreading in large wireless networks. 121

Figure 5.2 Spread of one-time failures. 122

Figure 5.3 Functions $a^2 \int_a^1 f(x)\mathrm{d}x$ and $(1-a^2)\int_0^a f(x)\mathrm{d}x$ when r is a constant, i.e., $f(x) = \delta(x-r)$. 125

Figure 5.4 Failure percolation with $p\lambda = 9.8$ and uniformly distributed $f(x)$. 126

Figure 5.5 Failure percolation with $p\lambda = 2.3$ and uniformly distributed $f(x)$. 127

Figure 5.6 Spread of persistent failures. 128

Figure 5.7 Mapping from continuum percolation to discrete percolation. 129

Figure 5.8 Failure percolation with $\lambda = 5$, $r = 1$, and $p = 0.38$. 135

Figure 5.9 Failure percolation with $\lambda = 5$, $r = 1$, and $p = 0.0025$. 135

Figure 5.10 Function $\Psi(g'(x), X_o, \Delta) \triangleq \int_\Delta g'(\|X_o - X_\varepsilon\|)\mathrm{d}\varepsilon$. 136

Figure 5.11 Failure percolation with $\lambda = 5$, $r' = 1$, and $g'(x) = 1 - 0.56x$. 138

Figure 5.12 Failure percolation with $\lambda = 5$, $r' = 1$, and $g'(x) = 0.0052 - 0.0052x$. 139

Chapter 1

Introduction

1.1 Motivation

In recent years, wireless networks attracted increasing attention from researchers because of their promising and indispensable usages in a wide range of communication scenarios. In places where the conventional wired networks are inconvenient, uneconomic or infeasible to be deployed, wireless networks provide an alternative or supplementary networking service. In fact, various types of wireless networks have already become an integral part of our daily lives, changing the traditional ways for communication in our many work and social activities.

The most widely observed example is the cellular network. Almost everyone in the world is benefiting from the communication convenience offered by this wireless cellular technology. Especially when the high-rate data transmission service is provided in the current 3G networks, the usage of and dependence on cellular networks are becoming unprecedentedly high. The deployment of cellular networks has enabled their users to have both voice communications and information exchanges anytime and anywhere in the areas covered by the networks.

On a smaller deployment scale, like the metropolitan area, wireless networks are also playing an increasingly advantageous role to meet our communication needs. The WiMAX technology allows the users to access information on the Internet using portable mobile broadband devices, providing an alternative solution to substitute the traditional cable and DSL connections. The high-speed Internet access offered by WiMAX provides a number of real-time services to the users including, for example, Voice-over-IP and IPTV.

In home and office areas, the wireless LAN and bluetooth technologies are presenting a much convenient data communication paradigm by removing the requirement of wired connections between different electronic devices. For example, the computers in an office are able to exchange files and access Internet resources through a wireless router, which provides the equivalent data transmission speed as the wired links. By using a bluetooth headphone, we can listen to music or have phone conversations when moving in the office.

In places that need to be monitored for security and protection reasons, wireless sensor networks are being studied for their potentials in deployment. When deployed, the sensor networks can provide automatic and continuous surveillance for the protected area. An example of the application is the forest fire detection. Using a sensor network to monitor the temperature change in a forest is a far more accurate and economic method than using human patrols, since sensors work days and nights without interruptions in all weather conditions.

The wireless network technologies have also made many traditional human tasks much easier to accomplish. For example, in the traditional way of reading electricity meters, the energy facility company needs to send workers to read the electricity meter of each apartment. The current research progress in the application of wireless network technologies is expected to substitute the human workers with the wireless meter reading devices that automatically and instantaneously acquire all the readings of a building.

The fast pace in the wireless network research and deployment is significantly changing our world in many aspects. The use of wireless networks is expected to be even more ubiquitous and pervasive in the future. New applications will be developed to satisfy the diversified user communication needs and various types of networks will be connected to create a large information fusion center to offer a magnitude of information services to the users, which will attract an ever increasing number of users into the wireless networks.

Accompanying the many advantageous usages of wireless networks, however, there exist a large number of technical challenges that require satisfactory solutions in order for us to make the full utilization of the wireless networks. In wireless networks, it is much more difficult to manage the network operations than in the wired networks. For example, the following tasks all require thoughtful solutions: radio resource allocation, transmission coordination, packet loss control, packet delay reduction, throughput maximization, user mobility support, failure prevention and resilience, user privacy and network security, interaction and inter-operability among networks, and service personalization and customization.

The limitation of radio frequency spectrum is one of the major constraints on the wireless network performance. As the wireless signals transmitted on the same frequency carrier interfer with one another, a wireless network can only use a small fraction of the frequency spectrum. Once a band of frequency spectrum is allocated to a wireless network, other wireless networks cannot utilize the same frequency band unless they are separated far away from one another. Given the limitation of the total available frequency spectrum, optimal allocation of the scarce radio resource is a research problem that requires deliberation and planning. On the other hand, the various real-time and peer-to-peer applications require an ever increasing bandwidth demand, which intensifies the radio resource scarcity problem further. Managing the allocated frequency spectrum efficiently to satisfy the service requirements is hence important in wireless networks.

Due to the communication openness, every packet transmitted in a wireless network is broadcast in nature. For all the non-intended nodes, a packet transmission in fact imposes an interference. When multiple nodes transmit concurrently in a contention to access the shared wireless medium, their transmitted signals may collide such that none of the transmitted packets is received correctly. The potential packet collisions hence require a coordination mechanism among the multiple senders if all have data to transmit. In order to meet the transmission requirement of every node, the coordination mechanism may need to collect the information of the link collisions and the node buffers to generate an optimal transmission schedule. It is a challenging research problem to collect the necessary information from the network and construct an efficient transmission schedule with moderate signaling overhead.

As the wireless links are subject to the dynamic physical environment, the quality of the transmitted signals on a wireless link is much lower than that of a wired link. When the transmissions of different nodes are not coordinated well or any part of the network topology changes, packets are lost in their transmissions. Detection and recovery of the lost packets are important in wireless networks due to the increased loss probability. There exist different strategies on the packet loss detection and recovery. For example, every intermediate relay node on the packet transportation path may be responsible for the hop by hop correct packet reception or the destination node undertakes the task of detecting packet losses and requesting packet retransmissions from the source node. Determination of the suitable strategy in a wireless network to control the packet loss is hence another research problem that requires a thoughtful study.

Compared to the wired links, the transmission capacity on wireless links is much limited due to the attenuation of transmitted signals and the high environmental noise and interference. Packets are transmitted with comparatively longer delays on wireless links. Besides, because of the increased bit error rate in the wireless settings, packets may not be received correctly. In case that a received packet cannot be decoded, the packet must be retransmitted until it is decoded correctly by the receiver. Every retransmission results in additional delay of the packet in the network. Furthermore, as the overall traffic carried in a wireless network increases, each packet also expects long delay on its path to the destination. Finally, as the network topology in wireless networks is dynamic, a packet may have to be buffered at an intermediate node before it can find a viable path to its destination. Hence, design of strategies to reduce the end-to-end delay in wireless networks is a non-trivial problem.

To utilize a wireless network to the fullest extent, we expect each packet in the network to be delivered to its destination in the shortest possible time and meanwhile every node to be able to send as much traffic as possible. As the traffic flows from different sources compete for the network resources, each node cannot take an arbitrarily large share of network resources for the transmission of its traffic, and hence its data sending rate must have a maximum bound. For instance, when a node sends, the other nodes within the interference region of the sending node cannot send. Otherwise, the packets from the interfering senders collide and none of the transmitted packets can be received by the intended receivers. In addition, when a packet travels through multiple relay nodes to reach its destination, it takes the transmission resources from each relay node in a competition with all the other traversing traffic. Under the constraint of limited network resources, throughput maximization is important and necessary.

Though the wireless networks have limited capacity as compared to the wired networks, they bring significant convenience to the users. An outstanding feature is the support to user mobility. Users of a wireless network are free to move in the network covered areas while maintaining their connections to other users. The mobility support is however a complicated process on the network side. The network needs to keep track of the user movement, allocate network resources to the user at its new location, and keep the user communications uninterrupted when in transit. When the traffic is transported in the network via multi-hop relays, the network must also manage the topology changes properly to minimize the impact of a moving node on the communication relay of other nodes.

Further, if all the nodes are movable in the network, the mobility management becomes even challenging. Effective solutions for the user mobility support in wireless networks are therefore another challenging research problem.

Like all other complicated systems, wireless networks have to deal with failures. Failures may occur in a wireless network due to a variety of reasons, for example, including the design errors, device malfunctions, environment changes, and power outages. Failures need to be considered at the time of network design and deployment to prevent them at the beginning, mitigate them at the occurrence, and eliminate them after the network restoration. In some cases, failures are random and independent from one another. In other cases, however, failures are correlated, which may result in a cascading sequence of many component failures in the network. To maintain a wireless network functional all the time, we need to investigate carefully the causes of various types of failures and take measures to minimize the possible impact created by failures in a wireless network. Given the complexity of wireless networks, failure prevention and resilience are quite complicated but important research problems.

User privacy and information security are also important in wireless networks. The information transmitted on wireless links is easily overheard by the non-intended users due to the broadcast nature, which may leak a user's private information to others. Malicious users may take unauthorized usage of other person's information, compromising the affected user's privacy. In other forms of attacks targeting the wireless networks and their users, the sophisticated attackers may even interrupt the normal communications of legitimate users, falsify the information transmitted by others, drop the data packets from other users, and jam the wireless channel to disconnect the users from the network. In fact, there has been an increasing attention from both the research community and the network industry to the potential threat on the wireless network security. Privacy and security concerns with wireless networks need to be addressed properly in order to provide trustworthy services to the users.

In practice, different wireless networks may co-exist to provide diversified services to the same group of users. For example, a wireless LAN and a bluetooth network may exist in the same office area. To enable flexible data exchange functionalities, we may want the devices on the two networks to be able to communicate to one another. The interaction involves some challenging research issues, for instance, the selection of a multi-homed device to serve as the gateway to connect two networks, the scheduling of node access to the wireless

networks to allow fair use of each user, and the control of traffic congestion at the gateway node. In other practical scenarios, wireless networks may be connected to the Internet, in which many inter-operability problems need to be solved satisfactorily. For example, we may want to treat the packet loss in such scenarios differently based on the specific causes. If packets are lost due to the congestion in the wired segment of transmission, a node should send slowly. On the contrary, if packets are lost due to the low signal quality in the wireless segment, a node should send aggressively to compensate for the packet loss.

Service personalization and customization are also interesting research problems in wireless networks. Users in a wireless network may have different service requirements that depend on the particular network usages of respective users. For example, a user who displays regular mobility and activity patterns may access the network at specific locations with specific requests, a user who is interested in certain news topics or technology developments may access specific network resources frequently, and a user who keeps communications with a group of social contacts may access specific types of social connection tools and services from time to time. Given the various network usage characteristics of different users, providing personalized and customized wireless network services to individual users or groups will be an attractive feature, which is however not an easy task due to a number of technical challenges including the estimation of network usage patterns and the design of user interested services.

On one hand, the deployment and utilization of wireless networks will continue increasing to offer pervasive and ubiquitous networking services. On the other hand, many technical problems still await satisfactory solutions in order to optimize the services. In this dissertation, we are dedicated to understand the performance and resilience of large-scale wireless networks, which lays the foundation for the further efforts in designing and operating large wireless networks in an efficient and resilient way. Specifically, we target two fundamental performance metrics, delay and throughput, and two network resilience aspects, mobility and failure, in this dissertation. Ideally, we expect a network to be operated with short delay and large capacity. Additionally, when mobility and failure occur, we also expect the network to continue working as normally as before.

The existing research results in the literature are however not sufficient for us to fully understand the best possible utilization that we may have in large-scale wireless networks. For example, the current studies on packet delay in large networks have mainly focused on the traffic load in the network. In these studies, the delay experienced by a

packet is determined by the waiting time in each relay hop before the packet is transmitted. However, in some application scenarios, such as sensor networks, the traffic load is usually not significant. Rather, due to the limited bandwidth and long transportation distance, the packet delay may be dominated by the transmission time spent on each link that the packet traverses. In this example, the existing results do not provide the answer for the fastest information propagation with light traffic load. In order to transport a message in the fastest speed, which could be urgent in some application scenarios, we need to re-consider the packet delay problem in large wireless networks.

Similarly, we observe an incomplete understanding of the communication capacity in large wireless networks. The existing studies have proven an upper bound on the capacity of large wireless networks. To achieve the maximum capacity, the transmissions in the network need to be coordinated carefully. All the existing solutions on the transmission scheduling employ a global control mechanism that assigns the transmission permission to each link and synchronizes all the concurrent transmissions. Such global mechanism is however not practical for large wireless networks. When the network size is large, collection of the network information regarding the node transmission requests and their interfering relations is difficult. Even if the required information is collected and a globally optimized transmission schedule is generated, the synchronization of transmissions is another challenge for the application of such a mechanism. Hence, we need to seek a practical scheme to maximize the capacity of large wireless networks.

The node mobility in wireless networks brings a concern on the stability of network structure. There are many studies in the literature that attempt to mitigate the impact of node mobility on the network architecture. For example, a hierarchical structure may be established to keep the topological framework intact when mobility occurs in local areas, the routing protocol may compare and select the most stable paths consisting of the low mobility nodes in the routing backbone, and a set of nodes may be used to bridge a connection between two nodes that are in and out of direct contact intermittently. Though the impact of node mobility on the network topology is alleviated by using these techniques, we do not know yet the possible extent of further stabilization in the network topology. We thus need to understand the wireless network topological stability with further insights, which will allow us to design optimized solutions to mitigate the impact of node mobility with improved effectiveness.

Failure resilience is another important research problem in large wireless networks.

The current studies on the failures in wireless networks have mainly focused on particular types of failures and their prevention and mitigation strategies. Though the existing work in the literature helps improve the wireless network resilience to certain types of failures, there lacks a general understanding of the failure impact on the overall wireless network structures and functions. In complex systems like a large wireless network, the countermeasures for single and random failures are usually taken into consideration in the phase of network design and deployment such that the network is quite resilient to isolated and occasional failures. However, in case that the failures are correlated to one another, the failure impact will be significantly larger than the case of isolated failures. Understanding the wireless network resilience to interdependent failures is hence necessary in order to prevent the network from potentially catastrophic failure impact.

Given the importance of these research problems, we are hence motivated to explore further the performance and resilience properties of large-scale wireless networks to gain a comprehensive understanding towards the maximal network utilization. We intend to determine the solutions for fast packet transportation, high traffic throughput, stable network structure and resilient counter-failure capability. Our work is helpful for the designers and operators of large-scale wireless networks to evaluate, improve and optimize their network usage.

1.2 Research Objectives

We will investigate four specific problems related to the performance and resilience of large-scale wireless networks, namely, the information propagation speed, the communication capacity, the topological stability with node mobility, and the network resilience against correlated failures. We next elaborate on these individual problems respectively.

1.2.1 Understanding the Information Propagation Speed

Packet delay has been a long-established topic in network research. It measures the time needed to transport a packet from its source node to its destination node. When quality-of-service is important for the packet, its delay usually must satisfy certain requirement. The delay experienced by a packet contains two components, the bandwidth-incurred delay and the load-incurred delay. Due to the limit of link bandwidth, it takes a packet some time to travel through a link, while traffic load introduces extra delay when the packet waits

for transmission. The existing works seen in literature all have focused on the load-incurred delay, where the statistical average of many packets [19–27] is investigated. In these studies, the delay is conditioned on the specific load in the network. In networks where the traffic load is low, the bandwidth-incurred delay becomes the dominant component, which is however not understood sufficiently yet. By determining the minimum bandwidth-incurred packet delay, we will gain the knowledge of the shortest possible packet delay in all traffic load cases, since traffic load incurs extra delays in addition to the bandwidth-incurred delay. Therefore, we are motivated to determine the minimum packet delay with arbitrary traffic load by considering the available bandwidth of transmission links and furthermore to find methods that achieve the fastest packet transportation in large wireless networks when traffic load is light.

In order to understand the fastest packet transportation in large-scale wireless networks, we define a new metric called the information propagation speed. The information propagation speed measures the geographical distance traveled by a packet within a given time period. We intend to achieve the following research objectives regarding this new network performance metric:

- The upper bound on information propagation speed will be determined, which reflects the minimum packet delay in a wireless network due to the constraint of link capacities.
- The conditions to achieve the maximum information propagation speed will be identified.
- In case that the ideal conditions are not satisfied in a wireless network, the gap between the upper bound and the actually achieved speed will be evaluated.
- A new packet transmission and relay scheme will be designed to minimize the packet delay in lightly-loaded networks.

We first determine the constraint on information propagation speed imposed by the network factors including node transmission powers, ambient and interference noises, and path selections. The theoretical understanding of speed bound will establish the goal for our subsequent efforts in minimizing packet delay. Through the mathematical analysis, we identify the necessary network conditions and configurations that enable us to achieve the fastest packet transportation. We also consider the cases in which the desired conditions

are not available. In these situations, we determine the gap between our achieved performance and the expected best performance in packet delay. Finally, we use the identified network conditions and configurations to design a simple routing scheme that expedites packet transportation and delivery. Compared to the pervious works on packet delay minimization, our work determines the fundamental delay bound applicable to arbitrary traffic load, and our routing scheme supplements the existing routing strategies in the light-loaded network scenarios. This study will maximize the network performance from the packet delay minimization perspective.

1.2.2 Understanding the Communication Capacity

Besides packet delay, network capacity is an equally important metric for network performance. It measures the total volume of packets that can be served simultaneously. When we consider all the packets inside a wireless network, the strategy that minimizes the delay of a specific packet may not maximize the network throughput, because the limited network resource may not be able to accommodate the same transportation service to all the packets. Many studies in the literature [1, 7, 12, 34–37, 48, 53] have investigated the capacity of large-scale wireless networks, in which methods have been proposed to achieve at least a constant fraction of the capacity bound. As part of the proposed capacity-achieving methods, all the current studies assume that the links in the entire network can be synchronized for collision-free transmissions, which is used in constructing joint routing and scheduling schemes to prove the achievability of order optimal network capacity. However, global link synchronization is very difficult to implement in large wireless networks. Hence, we are interested in determining whether the same order optimal capacity is still achievable in practical and realistic network settings in which links are synchronized only within local areas. If the answer is positive, we will be able to obtain the maximum network throughput in the order sense with significantly reduced implementation complexity and hence improved scheme practicality. We attempt to achieve the following research objectives:

- The capacity of large-scale wireless networks with localized link scheduling schemes will be determined, which is proven to be as optimal as the capacity achieved by using global link scheduling strategies.
- The governing conditions for achieving the same order optimal network capacity will be determined for localized link scheduling schemes.

- A distributed network partitioning protocol and a localized link scheduling algorithm will be designed to achieve the optimal network capacity with significantly reduced complexity and overhead.

As the first step toward designing realistic capacity-achieving schemes, we prove that the same order of network capacity is in fact achievable by using localized link scheduling only. Furthermore, we determine the network conditions and configurations that are both sufficient and necessary in order for localized link scheduling to achieve the order optimal network capacity. The identified conditions provide informational guidelines on the design of scheduling schemes, based on which we propose a distributed network decomposition protocol and a localized link scheduling scheme to maximize the network capacity in the sense of scaling order.

1.2.3 Understanding the Topological Stability with Node Mobility

Node mobility creates a significant impact to the stability of network structure. Due to the limited transmission power, a node may communicate only to the other nodes that are close enough within the reach of its transmission. When a node moves, its neighbors change over time. Since the moving node relies on its neighbors to relay its packets and the moving node may also be actively forwarding the packets from others, node movement results in frequent network topology changes and communication interruptions. When the broken communication links and paths are repaired, network overhead, packet delay and packet loss are incurred inevitably. The mobility impact is even greater in large-scale networks than the small ones because the average path length is longer in the former and any link break causes the path failure. We are hence motivated to stabilize the network topology in a mobile environment. Specifically, we construct a hierarchical network architecture and attempt to achieve the following research objectives:

- Through mathematical modeling of the random node mobility, the maximum topological stability will be determined in the hierarchical mobile wireless networks.
- A new node clustering algorithm will be designed to maximize the structural stability of clusters.
- A new routing protocol will be designed to maximize the topological stability of the communication links and paths.

The mathematical modeling and analysis lay the foundation for understanding and maximizing the topological stability in mobile wireless networks. To facilitate our study, we first choose a general node mobility model, which represents the typical node movements. Then we analyze the network topological stability in terms of the lifetimes of clusters, links and paths using the chosen node mobility model. The analytical results indicate the maximum stability of network topology in mobile environment and provide the guidelines for network design that maximizes the topological stability. Finally, we propose a new node clustering algorithm and a hierarchical routing protocol to stabilize the topology of mobile networks. Our study maximizes the network topology lifetime and avoids the excessive overhead associated with the topological changes incurred by node mobility.

1.2.4 Understanding the Network Resilience against Correlated Failures

Failures pose a significant threat to the correct network functioning. In large-scale wireless networks, node failures happen due to a variety of reasons including, for example, the device defects, battery depletions, and malicious alterations. When failures occur, not only the failed nodes suffer from the inability to communicate, but also the rest of the network may be impacted, as the failed nodes might serve as intermediate routers for the other portion of the network. Ideally, we want a wireless network to be robust such that it continues working when failures occur. In the scenario that the failures happen independently, we may ensure network resilience by deploying extra nodes in the network as backup. However, in some scenarios, failures display an interdependence in the way that an earlier failure might trigger the occurrence of follow-up failures, which exists as a complicated form of failure occurrence and a challenging problem for failure resilience. In order to design effective strategies to enhance the counter-failure capability of large-scale wireless networks, we must understand the spread of correlated failures first. Therefore, we target the following research objectives:

- A general mathematical model will be established to formulate the interdependent correlations among various types of failures.
- Based on the failure model, we will determine the correspondence between the degree of failure correlation and the extent of failure spread in large wireless networks.
- The analytical results on failure spread will be used as the foundation for designing

failure containment strategies to enhance the network resilience.

The first step in determining the scope of failure spread is to understand the logical connections between failures. We thus propose a mathematical framework to model the failure interdependence. Given the failure correlation model, we then determine the conditions on the correlation strength in which failures will and will not spread out to a significant portion of the network. Intuitively, stronger correlation implies higher chance of spreading. We attempt to reveal quantitatively the characteristic regimes of failure correlations for both the failure spreading occurrence and non-occurrence. Finally, we will suggest counter-failure strategies in large-scale wireless networks.

1.3 Contributions

Our work in this dissertation contributes toward a comprehensive understanding of the fundamental performance aspects of large-scale wireless networks. As we are unable to study exhaustively all the possible aspects of wireless networks, we have identified four of them in this research, namely, the information propagation speed, the communication capacity, the topological stability with node mobility, and the network resilience against correlated failures. Because these performance and resilience indicators reflect the fundamental expectations for large-scale wireless networks, we believe that our work provides useful insights for the network designers and operators to manage their wireless networks effectively, efficiently and reliably. More specifically, our research result on the information propagation speed determines the fastest message dissemination strategy, our result on the communication capacity offers a practical solution for maximizing the throughput in large wireless networks, our result on the topological stability extends the network architectural lifetime, and our result on the failure resilience provides a theoretical foundation to understand the impact of correlated failures on wireless networks. The analytical results and proposed solutions in this dissertation can be used to evaluate, improve and optimize the utilization of large-scale wireless networks.

1.4 Organization

The rest of this dissertation is organized as follows. Chapter 2 studies the maximum information propagation speed in large-scale wireless networks. Chapter 3 explores

the possibility and the approach for localized link scheduling to achieve the order optimal capacity in large-scale wireless networks. Chapter 4 presents our research result on the maximum topological stability of the hierarchical architecture in mobile wireless networks. Chapter 5 characterizes the spreading properties of correlated failures in large wireless networks. Finally, Chapter 6 summarizes our research results and discusses the possible extension directions.

Chapter 2

The Speed of Information Propagation in Wireless Networks

In this chapter we investigate the speed of information propagation in large wireless networks, which provides fundamental understanding of the fastest information transportation and delivery that a wireless network is able to accommodate. We show that there exists a unified speed upper bound for broadcast and unicast communications in large-scale wireless networks. When network connectivity is considered, this speed bound is a function of node density. If the network noise is constant, the bound is a constant when node density exceeds a threshold; if the network noise is an increasing function of node density, the bound decreases to zero when node density approaches infinity. As achieving the speed bound places strict requirements on node locations, we also quantify the gap between the actually achieved speed and the desired bound in random networks in which the relay nodes are not located as desired. We find that the gap converges to zero exponentially as node density increases to infinity.

2.1 Motivation and Related Work

Initiated by the seminal work of Gupta and Kumar [1], researchers have investigated the capacity of wireless networks thoroughly. It is found that the throughput per node decreases on the order of $\Theta\left(\frac{1}{\sqrt{n}}\right)$ as the node population n increases [1]. This finding is pessimistic, since it implies the fact that none of the nodes can communicate in the end if the network becomes overcrowded. In the succeeding efforts to improve capacity, it

is discovered that higher throughput is indeed obtainable if extra network conditions are considered, for example node mobility [2], infrastructure support [3], transmission cooperation [4] and ultrawide bandwidth [5]. Besides the theoretical bounds, algorithm design [6] has also been proposed to maximize the network capacity.

Different from the previous works, we study in this chapter *packet delay* instead of throughput of wireless networks. Packet delay is a performance metric equally important as throughput, especially in the Quality-of-Service sensitive real-time communications, in which the delay perceived by each packet is more pertinent to the communication quality than the throughput of the entire network. Specifically, we are interested in determining the lower bound on the packet delay in wireless networks. The delay bound provides the understanding of how fast a packet can be transported and paves the way for the further investigation on the feasibility of supporting delay-sensitive traffic in wireless networks.

The delay perceived by a packet is the combined result of various factors including path length, link bandwidth, traffic load, and channel access overhead. Fortunately, these factors can be decoupled as they represent different delay components. For instance, the path length determines the number of hops traversed by the packet; the link bandwidth determines the transmission time on each hop; and the traffic load comes into effect in the form of packet queuing, processing and wireless medium access delay at each intermediate node. In order to obtain generic results on packet delay, we do not assume any particular traffic pattern. Instead, we study a lightly-loaded network in which the packet queuing, processing and medium access delays are negligible as compared to the transmission delay. Decoupling the bandwidth-incurred delay and the load-incurred delay allows us to treat them separately and to reach conclusions that are not limited to specific traffic distributions. The additive property of packet delay guarantees that the lower bound discovered in the lightly-loaded networks also applies to the arbitrarily-loaded networks, though the tightness of the bound needs further investigation in the latter scenario.

To facilitate our study, we define the *Information Propagation Speed* as an equivalent metric of the packet delay. As packet transportation can be viewed as moving a packet from its source node to its destination node over a physical distance, we use the speed metric to quantify the movement progression, namely, the distance travelled through by the packet towards its destination in a given amount of time. Under this definition, we translate the original problem of finding the lower bound of packet delay into the equivalent problem of determining the upper bound of packet propagation speed. The fundamental trade-off is

the selection of few links with long distance per link versus the selection of many links with short distance per link. Obviously, short link distance improves the capacity of the link and therefore transmission finishes fast on the link, but a packet needs to go through many hops before reaching its destination. Zheng shows in [7] that there exists an upper bound W on the information propagation speed that is attainable under three conditions: i) every relay node uses an optimal transmission radius R, ii) the transportation distance between the source node and the destination node is a multiple of R, and iii) the relay nodes are aligned with equal separation distance R. We study in this chapter the generalized speed limit problem in which the above three conditions may not hold and, in addition, the network connectivity and the node location randomness are taken into account.

To be specific, we are interested in the following three questions. First, if the packet transportation distance between the source and the destination is known not to be a multiple of R, what is the upper bound on the information propagation speed? Obviously, it should be tighter than W since W is unreachable in this case. Second, does network connectivity place any constraint on the speed upper bound and, if yes, how? As achieving the maximum speed W requires using a specified transmission radius R, it is possible that the network is not fully connected by using R. Should this case occur, a packet may not be able to reach all the intended recipients and the speed upper bound W may not be feasible. Third, even if the optimal transmission radius R is used, it may not be possible to find the relay nodes at the desired locations due to the randomness in node distribution. When the relay nodes are not located as desired, information propagates at a lower speed and a gap exists between the actually achieved speed and the upper bound W. How to quantify this gap? We attempt to answer these questions, which provide the foundation for optimal network planning and protocol design to expedite packet delivery in multi-hop wireless networks.

As the first contribution, we show that the optimal transmission radius depends on the end-to-end transportation distance. In order to transport a packet at the fastest speed, the relay nodes must be equally spaced and use the same transmission radius. The value of the transmission radius is a divisor of the straightline distance between the source and the destination nodes. Interestingly, the optimal transmission radius converges to a constant in large-scale wireless networks.

As the second contribution, we determine the speed upper bound under the constraint of network connectivity. We introduce a probabilistic measurement on network

connectivity and examine the feasible speed bound in two noise models: the constant-interference model and the increasing-interference model. We show that in the constant-interference model the speed bound is a constant when node density exceeds a threshold and in the increasing-interference model the speed bound decreases to zero when node density increases to infinity.

As the third contribution, we quantify the gap between the actually achieved speed and the desired upper bound. The gap exists due to the randomness of node locations. We prove that a packet propagates omnidirectionally in large-scale wireless networks and the gap reduces as node density increases. We also show that in both noise models, there exists a threshold node density below which the gap is bounded by constants and above which the gap converges to zero exponentially.

Before we start to investigate the properties of information propagation speed in large-scale wireless networks, we first review the current efforts in the literature on both network capacity and packet delay.

Since the work by Gupta and Kumar [1], many efforts have been made to understand the fundamental performance limits of wireless networks, most of which have focused on the network throughput. Gupta and Kumar demonstrated that the throughput per node is $\Theta\big(\frac{1}{\sqrt{n}}\big)$ that decreases to zero as the number of nodes n goes to infinity. In the follow-up research, it is found that higher throughput can be achieved under various conditions and by using different techniques.

Grossglauser and Tse [2] discovered that mobility increases throughput. As the throughput of wireless networks is constrained by the interference of concurrent transmissions, minimizing interference improves throughput. Node mobility increases the chance of transporting a packet using a short path, which reduces the number of relay transmissions and alleviates the intensity of interference. Garetto *et al.* [8] further proved that the asymptotic capacity of ad hoc networks varies from $\Theta\big(\frac{1}{\sqrt{n}}\big)$ to $\Theta\big(1\big)$ under anisotropic mobility patterns.

Liu *et al.* [3] studied the wireless networks with infrastructure support. They found that throughput increases linearly with the number of base stations if there are sufficient base stations in the network. Kozat and Tassiulas [9] showed similar improvement that the throughput per node is $\Theta\big(\frac{1}{\log n}\big)$ if the number of access points is large enough such that each access point serves limited number of wireless nodes.

Gastpar and Vetterli [10] investigated the network coding techniques and found

that throughput can be improved by cooperation in transmissions. It is shown that the throughput per node pair scales as $\log n$ asymptotically when the nodes collaborate in transmissions. Ozgur *et al.* [4] extended the network scenario to multiple source-destination pairs. They showed that an almost linearly-scaling capacity $\Theta\big(n^{1-\epsilon}\big)$ can in fact be achieved by intelligent node cooperations and distributed multi-input multi-output (MIMO) communications.

Negi and Rajeswaran [11] proved that each wireless node can obtain a throughput bounded between $O\big((n\log n)^{\frac{\alpha-1}{2}}\big)$ and $\Omega\big(\frac{n^{(\alpha-1)/2}}{(\log n)^{(\alpha+1)/2}}\big)$ if infinite bandwidth is available. Zhang and Hou [5] further demonstrated that the throughput per node scales in fact as $\Theta\big(n^{\frac{\alpha-1}{2}}\big)$ if the bandwidth is unlimited and the Shannon channel capacity is attainable.

All the works discussed above enhance the network utilization by increasing the total volume of transported packets within a given time period. It was meanwhile discovered that by allowing some packet delay the network throughput can be further improved [12–14] and the node energy consumption can be reduced [15, 18]. However, because packet delay cannot be arbitrarily relaxed, for example the real-time communications (e.g., voice conversations) and the time-sensitive messages (e.g., urgent event reports in sensor networks) require timely delivery, mechanisms are needed to guarantee satisfactory packet delay in the delay-constrained communication scenarios.

Krunz and Kim [19] analyzed the packet delay distribution and discard rate in wireless networks, from which they determined the wireless effective bandwidth subject to the delay constraint. The computed effective bandwidth allows the enforcement of connection admission control. Wang *et al.* [20] proposed two admission control schemes to guarantee packet delay based on the statistical delay analysis.

Liebeherr *et al.* [21] and Verloop [22] studied packet scheduling policies to minimize the flow-level delays. In bandwidth-sharing networks, multiple classes of flows compete for the limited transmission bandwidth. Their competition is coordinated by a scheduling policy. Optimally configuring the scheduling policy not only achieves fairness among the flows but also minimizes the delay experienced by each flow.

Yang and Kravets [23] considered the wireless medium access delay. They analyzed the relationship between the delay of flows and the contention window sizes of nodes. Based on the analysis, they proposed a distributed delay allocation scheme that adjusts the contention window size of competing nodes to ensure that the average delay of individual flows meet their requirements. Similar MAC delay is also investigated in [24] and [25]. Bader and

Ekici [24] optimized the network throughput and delay by implementing interference-aware packet injection mechanisms that coordinate the packet waiting times at the competing nodes. Liang and Dong [25] took the approach of urgency ranking by scheduling the transmissions of the most urgent packets first.

Link scheduling was also studied to minimize the end-to-end packet delay by coordinating the transmission orders of wireless links. Djukic and Valaee [26] designed heuristics to select transmission order that minimizes the packet delay at each node incurred by synchronization mismatch between inbound and outbound links. Chafekar *et al.* [27] considered the concurrent packet transmissions between multiple source-destination pairs and presented a scheduling algorithm that minimizes the delivery delays of all the packets.

Though the existing research results on characterizing and minimizing packet delay have provided profound insights into the delay composition in wireless networks and valuable suggestions on the delay control methodologies, they all address the problem from a statistical perspective that minimizes the *average* delay of packet flows. In all these studies, traffic load is the essential factor contributing to packet delay. Quite differently, we consider in this chapter the delay of a *single* packet in lightly-loaded networks. As there are few other packets competing for the resources, the queuing and medium access delays are negligibly small. Instead, the end-to-end packet delay is dominated by the bandwidth of the links along the packet transportation path.

Till now, only Zheng [7] has studied this bandwidth-limited delay in wireless networks. In the pioneering paper [7], Zheng defined the concept of *Information Diffusion Rate* that measures the time required for disseminating a packet to every node in the network. It is shown that a packet cannot be disseminated faster than a constant upper bound. The fastest rate is achieved only if the packet travels through a straightline path consisting of R-distance equally spaced relay nodes and the farthest node in the network is a multiple of R-distance away from the source, where R is a constant.

2.2 Problem Formulation

We study the generalized packet propagation speed problem in this chapter. Particularly, we attempt to discover the speed bound for transporting a packet between arbitrarily located source-destination pairs and take into consideration the network connectivity requirement as well as the node location randomness. As mentioned earlier, we assume a

lightly-loaded network where the packet delay and hence the packet propagation speed are solely determined by the link bandwidth. The speed upper bound is, nonetheless, still valid for networks with any load distributions. Before starting the investigation, we first describe our network model and define the metric *Information Propagation Speed*.

2.2.1 Reference Network Model

We study a square-shaped wireless network of n nodes in a very large area $\mathcal{B} = [-\frac{l}{2}, \frac{l}{2}]^2$ ($l \to \infty$) with the following assumptions regarding the node locations and communications.

- The nodes are static and randomly distributed obeying a Poisson point process [28] with density λ.
- All the nodes share a B Hz available frequency band.
- Any two nodes can communicate over the direct link between them. The link is characterized by an Additive White Gaussian Noise (AWGN) channel with path loss exponent $\alpha \geq 2$ [29] and its bandwidth is subject to the *Shannon Limit* [30]: $C = B\log_2(1+\mathrm{SNR})$, where SNR is the signal-to-noise ratio. Advanced coding techniques are used such that the link bandwidth approximates the Shannon limit [31].
- A uniform transmission power P is used by every node.
- The noise N present at each node is the sum of the ambient noise N_A and the interference noise N_I: $N = N_A + N_I$. For tractable modeling and analysis, we assume that N_A is a constant and N_I is either a constant or a variable, depending on the density of concurrently transmitting nodes. We consider two cases: i) the *constant-interference* noise model in which N_I is a constant when λ increases and ii) the *increasing-interference* noise model in which N_I increases as λ increases. In the second model, since N_I dominates in the total noise when λ increases, we ignore N_A and assume $N = N_I$. In both models, the total noise N is assumed to be homogeneous in the network.
- No directional antenna is used and no large signal-blocking obstacle exists in the network.
- A packet has fixed length of L bits during transportation.

For the purpose of clearer presentation, we make the following comments and clarifications on the assumptions stated above.

- Throughout this chapter, the area of the wireless network $|\mathcal{B}|$ is fixed, though it can be arbitrarily large. When we study the limit case of infinite node density ($\lambda \to \infty$), the density scales by increasing the node population n while keeping the area $|\mathcal{B}|$ constant.

- All the distances in this chapter are the Euclidean distance.

- The path loss attenuation exponent α is 2 in free space. In all other environments, α is bigger than 2 due to multipath fading effects.

- Given two nodes v_i and v_j separated by a distance $d_{v_iv_j}$, the signal strength of v_i as received by v_j is $Pd_{v_iv_j}^{-\alpha}$. Hence, the capacity of the direct link between v_i and v_j is $C_{v_iv_j} = B\log_2(1 + \frac{P}{N}d_{v_iv_j}^{-\alpha})$.

- If v_i sends a packet to v_j at the full bit rate $C_{v_iv_j}$, a node v_k also receives the same packet if $d_{v_iv_k} \leq d_{v_iv_j}$, since $C_{v_iv_k} \geq C_{v_iv_j}$. On the other hand, if $d_{v_iv_k} > d_{v_iv_j}$, v_k cannot receive the packet correctly, as $C_{v_iv_k} < C_{v_iv_j}$. We define $r_{v_i} = \max_s\{d_{v_is}|C_{v_is} \geq C_{v_iv_j}\}$ as the *transmission radius* of v_i and $\mathcal{A}_{v_i} = \{s|d_{v_is} \leq r_{v_i}, s \in \mathcal{B}\}$ as the *coverage area* of v_i in this transmission.

2.2.2 Information Propagation Speed

In multihop wireless networks, the transportation of a packet is via rebroadcasting. As illustrated in Fig. 2.1, when node v_0 initiates a packet transportation, it broadcasts the packet to all the neighbors inside its coverage area $\mathcal{A}_{v_0}$ and these neighbors continue to rebroadcast the packet to farther distance until the packet is received by the destination node. Depending on the routing protocol used, not every intermediate node that hears the packet is required to rebroadcast. Besides, node scheduling is often implemented in large wireless networks to separate the simultaneous transmissions such that their packets do not collide. From the perspective of information theory, the interference from simultaneous transmissions only degrades the quality of wireless channels but not necessarily precludes communications. Therefore, theoretically speaking, communications are still possible without node scheduling. However, in order to be consistent with the de facto practice, we

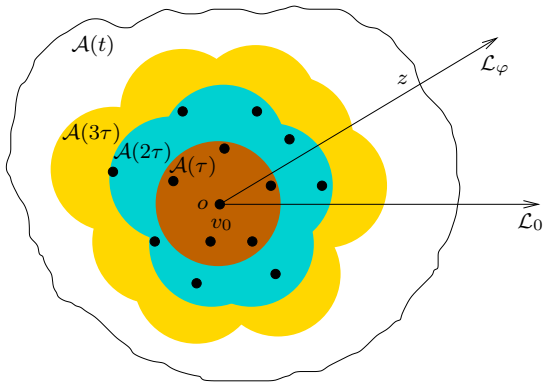


Figure 2.1: Information propagation in multihop wireless networks. The packet broadcasting duration is τ for each node.

assume that a random η percentage of nodes are scheduled for transmission at any time. Thus, considering packet routing and node scheduling, only a subset of the nodes that have received the packet from v_0 rebroadcast to transport the packet.

Denote $\mathcal{V}(t)$ as the set of nodes that have received the packet by time t and $\widetilde{\mathcal{V}}(t) \subset \mathcal{V}(t)$ as the subset that have forwarded the packet by time t. The total area that the packet has reached by time t is then expressed as $\mathcal{A}(t) = \cup_{v_i \in \widetilde{\mathcal{V}}(t)} \mathcal{A}_{v_i}$. In addition, denote $\mathcal{L}_\varphi$ as the line starting from v_0 in the direction $\varphi \in [0, 2\pi)$ and define $\mathcal{L}_\varphi(t) = \mathcal{L}_\varphi \cap \mathcal{A}(t)$. In Fig. 2.1, $\mathcal{L}_\varphi(t)$ is the line segment $\overline{oz}$. The *Information Propagation Speed* in the direction φ is then defined to be

$$w_\varphi(t) = \frac{|\mathcal{L}_\varphi(t)|}{t} = \frac{\max_s\{d_{v_0 s} | s \in \mathcal{L}_\varphi(t)\}}{t}. \tag{2.1}$$

By definition, $w_\varphi(t)$ is the distance from v_0 to the farthest location reached by the packet in direction φ divided by the time elapsed since the packet departure from v_0. As we will show in the rest of this chapter, maximizing $w_\varphi(t)$ requires all the packet relay nodes use a specified transmission radius. We name such a radius as *optimal transmission radius* in the sense of $w_\varphi(t)$ maximization.

We will next formally derive the upper bound on $w_\varphi(t)$, examine the feasibility of the upper bound under the network connectivity constraint, and determine the gap between $w_\varphi(t)$ and its upper bound.

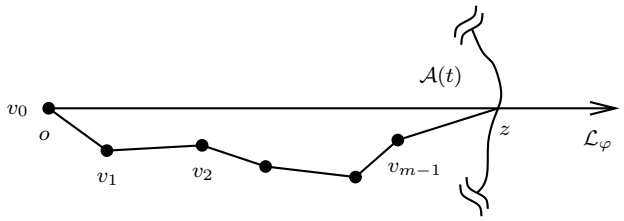


Figure 2.2: The packet relay path in direction φ.

2.3 The Speed Upper Bound

Zheng shows in [7] that there exists a constant upper bound on $w_\varphi(t)$ that is attainable when several conditions are satisfied simultaneously. One of the conditions requires that the source and the destination be separated by a distance that is multiple of the optimal transmission radius R, where R is a constant independent of the source-destination distance. We note that in broadcast communications R might be the best transmission strategy for fast packet dissemination, since the number of destinations may be large and their locations may not be known. However, in unicast communications there is only one destination, the location of which is possibly known to the source and relay nodes. If the known source-destination distance is not a multiple of R, we show that there exists a tighter speed upper bound that is achieved at a different transmission radius. For completeness of presentation, we first reproduce the result of [7], as shown next in the case of broadcast communications.

2.3.1 Bound in Broadcast Communications

Suppose by time t a packet originated from v_0 has reached the location z in direction φ, as shown in Fig. 2.2. Let $\mathcal{P} = \{v_0, v_1, \cdots, v_{m-1}\}$ denote the relay path from o to z and $\tau_{v_i} = \frac{L}{B\log_2(1+\frac{P}{N}r_{v_i}^{-\alpha})}$ denote the transmission duration of node v_i. By definition,

$$\begin{aligned} w_\varphi(t) &\leq \frac{\sum_{i=0}^{m-1} r_{v_i}}{\sum_{i=0}^{m-1} \tau_{v_i}} \\ &\leq \max_i \frac{r_{v_i}}{\tau_{v_i}} \\ &\leq \frac{B}{L} \max_r r\log_2(1+\frac{P}{N}r^{-\alpha}). \end{aligned} \tag{2.2}$$

The maximum of $r\log_2(1+\frac{P}{N}r^{-\alpha})$ occurs when $r = R_b = (\frac{P}{Ny(\alpha)})^{\frac{1}{\alpha}}$, where $y(\alpha)$ is the non-zero root of the equation

$$(1+y)\log_2(1+y) = \frac{\alpha}{\ln 2}y. \tag{2.3}$$

Thus,

$$w_\varphi(t) \le W_b = \frac{B}{L}R_b\log_2(1+\frac{P}{N}R_b^{-\alpha}). \tag{2.4}$$

If B, L, P, N and α are constant, R_b and W_b are constant. This is the same result as Zheng has obtained in [7]. Inequality (2.4) demonstrates that $w_\varphi(t)$ is upper bounded by the constant W_b, which is achieved when each step in (2.2) takes equality, i.e., the following conditions are satisfied: i) every relay node uses the optimal transmission radius R_b, ii) relay nodes are lined up and equispaced by R_b, and iii) the distance from v_0 to the destination node v_d (or the farthest recipient node in broadcast communications) is a multiple of R_b.

2.3.2 Bound in Unicast Communications

In unicast communications, though we can always require every node transmit in the radius R_b, the distance $d_{v_0v_d}$ between the source and the destination nodes may be known and not equal to a multiple of R_b. Should this case occur, we show next that $w_\varphi(t)$ is upper bounded more tightly by another constant W_u that is achievable when a different transmission radius R_u is used, as specified in the following theorem.

***Theorem* 2.1.** *$w_\varphi(t)$ is maximized when the optimal transmission radius R_u is determined by:*

$$R_u = \begin{cases} d_{v_0v_d} & \text{if } d_{v_0v_d} < d^*, \\ d_{v_0v_d}/\mathrm{G}(d_{v_0v_d}/R_b) & \text{if } d_{v_0v_d} > d^*, \end{cases} \tag{2.5}$$

in which $d^* = (\frac{P}{N(2^\alpha-2)})^{\frac{1}{\alpha}}$ and function $\mathrm{G}(\cdot)$ rounds $d_{v_0v_d}/R_b$ to the nearest integer. In both cases, $w_\varphi(t) \le W_u = \frac{B}{L}R_u\log_2(1+\frac{P}{N}R_u^{-\alpha})$.

***Remark* 2.1.** *Theorem 2.1 states the fact that there exists a threshold distance d^* such that: i) if the source-destination distance is shorter than d^*, direct transmission from v_0 to v_d achieves the fastest speed, and ii) if the source-destination distance is longer than d^*, the fastest speed is achieved when the optimal transmission radius takes the value closest to R_b and dividing the source-destination distance.*

In order to prove Theorem 2.1, we introduce a few notations and lemmas. Denote $\mathcal{P}^m(v_0, v_d) = \{v_0, v_1, \cdots, v_{m-1}, v_d\}$ $(m \geq 1)$ as an m-hop straightline relay path from v_0 to v_d, and $T(\mathcal{P}^m(v_0, v_d))$ as the packet transmission time along $\mathcal{P}^m(v_0, v_d)$. We have the following lemmas.

***Lemma* 2.1.** *Consider $\mathcal{P}^1(v_0, v_d)$ and $\mathcal{P}^2(v_0, v_d)$. Defining $\min t_{v_0v_d} = \min\{T(\mathcal{P}^1(v_0, v_d)), T(\mathcal{P}^2(v_0, v_d))\}$, we have*

$$\min t_{v_0v_d} = \begin{cases} \frac{L}{B\log_2(1+\frac{P}{N}d_{v_0v_d}^{-\alpha})} & \text{if } d_{v_0v_d} < d^*, \\ \frac{2L}{B\log_2(1+\frac{P}{N}(\frac{d_{v_0v_d}}{2})^{-\alpha})} & \text{if } d_{v_0v_d} > d^*. \end{cases} \tag{2.6}$$

***Remark* 2.2.** *Lemma 2.1 states the fact that: i) if $d_{v_0v_d} < d^*$, 1-hop direct transmission is faster than any 2-hop relay transmission; ii) if $d_{v_0v_d} > d^*$, choosing a relay node equidistant from the source and the destination results in the fastest transmission among all the 2-hop relay paths, and it is also faster than the 1-hop direct transmission; iii) if $d_{v_0v_d} = d^*$, 1-hop direct transmission is as fast as the 2-hop relay transmission with the relay node placed exactly at the middle, and both are faster than any other 2-hop transmissions.*

Proof. Define $t(x) = \frac{L}{B\log_2(1+\frac{P}{N}x^{-\alpha})}$ and $t_{v_0v_d}(x) = t(x) + t(d_{v_0v_d} - x)$, in which $x \in [0, d_{v_0v_d}]$ is the distance between the source node and the relay node. When $x = 0$ or $x = d_{v_0v_d}$, it is a 1-hop transmission. Otherwise, it is a 2-hop transmission. In order to find the minimum of $t_{v_0v_d}(x)$, we determine the convexity of $t(x)$ and $t_{v_0v_d}(x)$ first.

By definition, function $t(x)$ has these properties:

$$t'(x) = \frac{(\ln 2)\alpha LPx^{-\alpha-1}}{BN(1+\frac{P}{N}x^{-\alpha})\ln^2(1+\frac{P}{N}x^{-\alpha})} > 0,$$

$$\begin{aligned} t''(x) &= \frac{\alpha LPx^{-\alpha-2}}{(\ln 2)^2 BN(1+\frac{P}{N}x^{-\alpha})^2\log_2^3(1+\frac{P}{N}x^{-\alpha})} \\ &\quad \cdot[2\alpha\frac{P}{N}x^{-\alpha} - (\alpha+1+\frac{P}{N}x^{-\alpha})\ln(1+\frac{P}{N}x^{-\alpha})]. \end{aligned}$$

Define $y = \frac{P}{N}x^{-\alpha}$ and $f(y) = 2\alpha y - (\alpha+1+y)\ln(1+y)$. We have $f'(y) = 2\alpha - [\ln(1+y) + \frac{\alpha}{1+y} + 1]$. It is not difficult to find the following properties of $f'(y)$:

i) $f'(0) = \alpha - 1 > 0$,

ii) $f'(y)$ increases monotonically when $y \in [0, \alpha - 1)$,

iii) $f'(y)$ decreases monotonically when $y \in (\alpha - 1, \infty)$,

iv) $\lim_{y\to\infty} f'(y) = -\infty$.

These properties indicate $\exists\ y_1 > 0$ s.t.

$$\begin{cases} f'(y) > 0 & 0 \le y < y_1, \\ f'(y) < 0 & y_1 < y < \infty. \end{cases}$$

Since $f(0) = 0$, $\exists\ y_2 > y_1$ s.t.

$$\begin{cases} f(y) > 0 & 0 < y < y_2, \\ f(y) < 0 & y_2 < y < \infty. \end{cases}$$

Define $x_2 = (\frac{P}{Ny_2})^{\frac{1}{\alpha}}$ and $f(x) = 2\alpha\frac{P}{N}x^{-\alpha} - (\alpha + 1 + \frac{P}{N}x^{-\alpha})\ln(1 + \frac{P}{N}x^{-\alpha})$. Then,

$$\begin{cases} f(x) < 0 & 0 < x < x_2, \\ f(x) > 0 & x_2 < x < \infty. \end{cases}$$

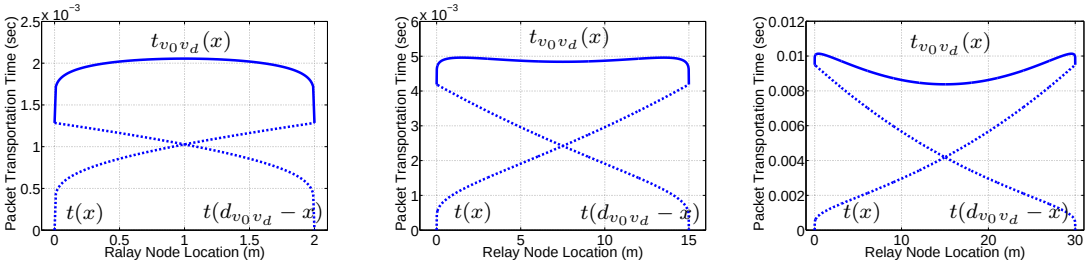
Since $\forall x > 0$, $\frac{\alpha LPx^{-\alpha-2}}{(\ln 2)^2 BN(1+\frac{P}{N}x^{-\alpha})^2 \log_2^3(1+\frac{P}{N}x^{-\alpha})} > 0$, we have

$$\begin{cases} t''(x) < 0 \text{ (i.e., } t(x) \text{ is strictly concave)} & 0 < x < x_2, \\ t''(x) > 0 \text{ (i.e., } t(x) \text{ is strictly convex)} & x_2 < x < \infty. \end{cases}$$

Next, we determine the convexity of $t_{v_0v_d}(x)$. Note that $t_{v_0v_d}(x)$ is symmetric with respect to $x = \frac{d_{v_0v_d}}{2}$. In addition, $t'_{v_0v_d}(\frac{d_{v_0v_d}}{2}) = 0$ and $t''_{v_0v_d}(0) = t''_{v_0v_d}(d_{v_0v_d}) < 0$ (because $t''(0) = -\infty$, $t''(d_{v_0v_d}) < \infty$, $t''_{v_0v_d}(0) = t''_{v_0v_d}(d_{v_0v_d}) = t''(0) + t''(d_{v_0v_d}) = -\infty$). The convexity as well as the minimum of $t_{v_0v_d}(x)$ is discussed in three cases.

i) $d_{v_0v_d} \in (0, x_2]$. $t(x)$ and $t(d_{v_0v_d} - x)$ are concave on $[0, d_{v_0v_d}]$, so $t_{v_0v_d}(x)$ is concave with no local minimum, as shown in Fig. 2.3(a).

ii) $d_{v_0v_d} \in (x_2, 2x_2]$. $t(x)$ is concave on $[0, x_2]$ and convex on $[x_2, d_{v_0v_d}]$. $t(d_{v_0v_d} - x)$ is convex on $[0, d_{v_0v_d} - x_2]$ and concave on $[d_{v_0v_d} - x_2, d_{v_0v_d}]$. So, $t_{v_0v_d}(x)$ must be concave on $[d_{v_0v_d} - x_2, x_2]$, while either concave or convex on $[0, d_{v_0v_d} - x_2] \cup [x_2, d_{v_0v_d}]$. However, $t''_{v_0v_d}(0) = t''_{v_0v_d}(d_{v_0v_d}) < 0$ indicates concavity. Thus, $t_{v_0v_d}(x)$ is concave on $[0, d_{v_0v_d}]$ with no local minimum, also as shown in Fig. 2.3(a).



(a) $0 < d_{v_0v_d} = 2\text{m} \le 2x_2$ (b) $2x_2 < d_{v_0v_d} = 15\text{m} < \infty$ (c) $2x_2 < d_{v_0v_d} = 30\text{m} < \infty$

Figure 2.3: The packet transportation time via 1-hop and 2-hop relay paths, $B = 100$ KHz, $L = 1024$ bits, $\frac{P}{N} = 10^3$, $\alpha = 2$.

iii) $d_{v_0v_d} \in (2x_2, \infty)$. Similar to the discussion in case ii), $t_{v_0v_d}(x)$ must be convex on $[x_2, d_{v_0v_d} - x_2]$, while either concave or convex on $[0, x_2] \cup [d_{v_0v_d} - x_2, d_{v_0v_d}]$. Again, $t''_{v_0v_d}(0) = t''_{v_0v_d}(d_{v_0v_d}) < 0$ indicates concavity. So, $t_{v_0v_d}(x)$ has one local minimum at $x = \frac{d_{v_0v_d}}{2}$ and two local maxima in $[0, x_2]$ and $[d_{v_0v_d} - x_2, d_{v_0v_d}]$, as shown in Fig. 2.3(b) and 2.3(c).

Summarizing all the three cases, $t_{v_0v_d}(x)$ has at most one local minimum which occurs at $x = \frac{d_{v_0v_d}}{2}$. Hence,

$$\begin{aligned} \min t_{v_0v_d} &= \min\{t_{v_0v_d}(0), t_{v_0v_d}(\frac{d_{v_0v_d}}{2}), t_{v_0v_d}(d_{v_0v_d})\} \\ &= \begin{cases} \frac{L}{B\log_2(1+\frac{P}{N}d_{v_0v_d}^{-\alpha})} & \text{if } d_{v_0v_d} < d^*, \\ \frac{2L}{B\log_2(1+\frac{P}{N}(\frac{d_{v_0v_d}}{2})^{-\alpha})} & \text{if } d_{v_0v_d} > d^*. \end{cases} \end{aligned}$$

□

Next, we generalize the result of Lemma 2.1 to the cases of any-hop path length and introduce Lemma 2.2.

Lemma 2.2. *Consider* $\mathcal{P}^m(v_0, v_d)$ $(m \ge 1)$ *and define* $\min t_{v_0v_d} = \min_m\{T(\mathcal{P}^m(v_0, v_d))\}$.

$$\min t_{v_0v_d} = \begin{cases} T(\mathcal{P}^1(v_0, v_d)) & \text{if } d_{v_0v_d} < d^*, \\ \min_m\{T(\mathcal{P}_e^m(v_0, v_d))\} & \text{if } d_{v_0v_d} > d^*, \end{cases} \tag{2.7}$$

in which $\mathcal{P}_e^m(v_0, v_d)$ *is an* m*-hop equidistant relay path connecting* v_0 *and* v_d*, i.e.,* $\mathcal{P}_e^m(v_0, v_d) = \{v_0, v_1, \cdots, v_{m-1}, v_d\}$ *and* $d_{v_0v_1} = \cdots = d_{v_{m-1}v_d}$.

***Remark* 2.3.** *Lemma 2.2 states the fact that: i) if $d_{v_0v_d} < d^*$, 1-hop direct transmission is faster than any multihop transmissions; ii) if $d_{v_0v_d} > d^*$, the fastest transmission must be achieved along a relay path in which the relay nodes are separated equally.*

Proof. By Lemma 2.1, if $d_{v_0v_d} < d^*$, $T(\mathcal{P}^1(v_0, v_d)) < T(\mathcal{P}^2(v_0, v_d))$. $\forall\ m \geq 2$, as $d_{v_{m-1}v_d} < d_{v_{m-2}v_d} < \cdots < d_{v_0v_d} < d^*$, apply the result of Lemma 2.1 recursively,

$$\begin{aligned} T(\mathcal{P}^1(v_0, v_d)) &< \sum_{i=0}^{0} T(\mathcal{P}^1(v_i, v_{i+1})) + T(\mathcal{P}^1(v_1, v_d)) \\ &< \sum_{i=0}^{1} T(\mathcal{P}^1(v_i, v_{i+1})) + T(\mathcal{P}^1(v_2, v_d)) \\ &\vdots \\ &< \sum_{i=0}^{m-2} T(\mathcal{P}^1(v_i,v_{i+1})) + T(\mathcal{P}^1(v_{m-1},v_d)) \\ &= T(\mathcal{P}^m(v_0, v_d)). \end{aligned}$$

Therefore, $\min t_{v_0v_d} = T(\mathcal{P}^1(v_0, v_d))$ when $d_{v_0v_d} < d^*$.

In order to prove $\min t_{v_0v_d} = \min_m\{T(\mathcal{P}_e^m(v_0, v_d))\}$ when $d_{v_0v_d} > d^*$, it is equivalent to show that for any path $\mathcal{P}^m(v_0, v_d)$, there is another path $\mathcal{P}_e^{m'}(v_0, v_d)$ that satisfies $T(\mathcal{P}_e^{m'}(v_0, v_d)) \leq T(\mathcal{P}^m(v_0, v_d))$. We consider the following node removal and relocation process to prove the existence of $\mathcal{P}_e^{m'}(v_0, v_d)$. For each node $v_i \in \mathcal{P}^m(v_0, v_d)$, we make the two changes below in sequence.

1. **Node removal**. Find the set of nodes $\{v_j \mid d_{v_iv_j} \leq d^*, j = i+1, \cdots, i+k\}$. If $k > 1$, remove the nodes $\{v_j, j = i+1, \cdots, i+k-1\}$ from $\mathcal{P}^m(v_0, v_d)$.

2. **Node relocation**. If v_i is the last relay node or $v_i = v_d$, skip this step. Otherwise, if $k = 0$ (k is the number of nodes found in step 1), relocate v_{i+1} such that $d_{v_iv_{i+1}} = d_{v_{i+1}v_{i+2}}$; if $k > 0$, relocate v_{i+k} such that $d_{v_iv_{i+k}} = d_{v_{i+k}v_{i+k+1}}$.

This process initiates at v_0, proceeds node by node toward v_d, and iterates until there is no more node removal and no more node relocation in the resulting relay path $\mathcal{P}_e^{m'}(v_0, v_d)$. Fig. 2.4 depicts an example of these two procedures.

First, we show that the resulting relay path has shorter transmission time than the original path, i.e., $T(\mathcal{P}_e^{m'}(v_0, v_d)) \leq T(\mathcal{P}^m(v_0, v_d))$. In the node removal step, since $d_{v_iv_{i+k}} \leq d^*$, by the first part of Lemma 2.2 (already proven), 1-hop direct transmission

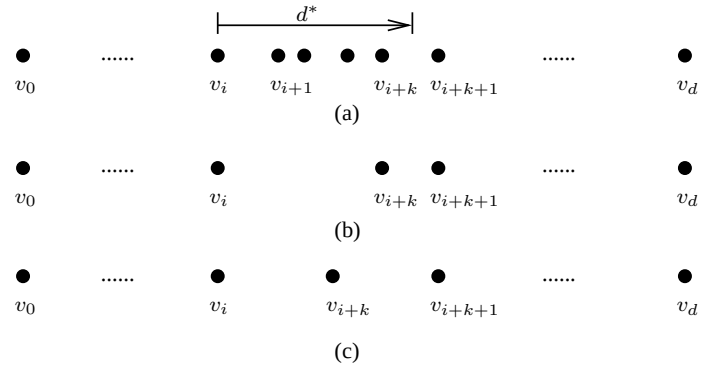


Figure 2.4: The node removal and relocation process. (a) Finding the node set $\{v_j \mid d_{v_i v_j} \leq d^*, j = i+1, \cdots, i+k\}$. (b) Removing the nodes $\{v_j, j = i+1, \cdots, i+k-1\}$. (c) Relocating the node v_{i+k} such that $d_{v_i v_{i+k}} = d_{v_{i+k} v_{i+k+1}}$ (if $k > 0$) or the node v_{i+1} such that $d_{v_i v_{i+1}} = d_{v_{i+1} v_{i+2}}$ (if $k = 0$).

from v_i to v_{i+k} is faster than the k-hop transmission via $v_{i+1}, \cdots, v_{i+k-1}$. Hence, removing $\{v_j, j = i+1, \cdots, i+k-1\}$ results in faster transmission. In the node relocation step, because $d_{v_i v_{i+2}}$ (if $k = 0$) or $d_{v_i v_{i+k+1}}$ (if $k > 0$) is larger than d^*, by Lemma 2.1, relocation of v_{i+1} (if $k = 0$) or v_{i+k} (if $k > 0$) results in faster transmission. Therefore, $\mathcal{P}_e^{m'}(v_0, v_d)$ has shorter transmission time than $\mathcal{P}^m(v_0, v_d)$.

Second, we prove that the resulting relay path $\mathcal{P}_e^{m'}(v_0, v_d)$ is a equidistant relay path, i.e., $d_{v_0 v_1} = \cdots = d_{v_{m'-1} v_d}$. Because the node removal step takes relay nodes away and the number of remaining relay nodes must be non-negative, it is obvious that the number of relay nodes converges to a value m' ($0 \leq m' \leq m$). After that, there are no more removals, but relocations may continue. As the transmission time from v_0 to v_d decreases during the relocations (already proven) and it is non-negative, it must converge to certain value, after which there are no more relocations. If $d_{v_0 v_1}, d_{v_1 v_2}, \cdots, d_{v_{m'-1} v_d}$ are not all equal, relocation will continue. Thus, they must be all equal by the end of the relocation process.

Finally, it is safe to substitute m for m'. After the substitution, we have $\min t_{v_0 v_d} = \min_m \{T(\mathcal{P}_e^m(v_0, v_d))\}$. □

We are now ready to prove Theorem 2.1 based on Lemma 2.2.

Proof. First, if $d_{v_0 v_d} < d^*$, Lemma 2.2 states that 1-hop direct transmission is the fastest among all the possible relay transmissions. It is obvious to conclude that in order to maximize $w_\varphi(t)$, $R_u = d_{v_0 v_d}$.

Second, if $d_{v_0v_d} > d^*$, Lemma 2.2 shows that $\min t_{v_0v_d} = \min_m\{T(\mathcal{P}_e^m(v_0, v_d))\}$. Since

$$T(\mathcal{P}_e^m(v_0, v_d)) = \frac{mL}{B\log_2(1 + \frac{P}{N}(\frac{d_{v_0v_d}}{m})^{-\alpha})},$$

we solve $\mathrm{d}T(\mathcal{P}_e^m(v_0, v_d))/\mathrm{d}m = 0$ for the optimal number of relay hops $m = \mathrm{G}(d_{v_0v_d}/R_b)$, the optimal transmission radius $R_u = d_{v_0v_d}/\mathrm{G}(d_{v_0v_d}/R_b)$, and the upper bound on the information propagation speed

$$w_\varphi(t) \le W_u = \frac{B}{L}R_u\log_2(1 + \frac{P}{N}R_u^{-\alpha}). \tag{2.8}$$

Because R_b is the unique maximizer for $r\log_2(1 + \frac{P}{N}r^{-\alpha})$, $W_u \le W_b$. When $d_{v_0v_d}$ is a multiple of R_b, $R_u = R_b$ and $W_u = W_b$. Otherwise, $R_u \ne R_b$ and $W_u < W_b$. □

We plot the packet transportation time for different source-destination distances in Fig. 2.5 as a visual aid to understand Theorem 2.1. Unlike R_b and W_b, R_u and W_u are determined not only by B, L, P, N and α, but also by the source-destination distance $d_{v_0v_d}$. The conditions for $w_\varphi(t) = W_u$ are: i) every relay node uses the optimal transmission radius R_u, and ii) the relay nodes are aligned and separated from each other by distance R_u. Note $\lim_{d_{v_0v_d}\to\infty} R_u = R_b$, indicating in large-scale networks the generalized speed bound for arbitrarily located source and destination nodes converges to the same constant bound for broadcast communications. As we study large-scale networks in this chapter, we will denote $R_b = R_u = R$ and $W_b = W_u = W = \frac{B}{L}R\log_2(1 + \frac{P}{N}R^{-\alpha})$ in the rest of this chapter, where $R = (\frac{P}{Ny(\alpha)})^{\frac{1}{\alpha}}$.

2.3.3 Solution of Optimal Transmission Radius

Till now, we have shown the existence of a unified optimal transmission radius R in large-scale networks. Next, we provide a solution for R. Equation (2.3) can be rewritten as

$$y = e^{\frac{\alpha y}{1+y}} - 1, \tag{2.9}$$

which allows us to compute $y(\alpha)$ in a recursive way:

$$\begin{cases} y_0(\alpha) = e^\alpha - 1, \\ y_i(\alpha) = e^{\frac{\alpha y_{i-1}(\alpha)}{1+y_{i-1}(\alpha)}} - 1 \quad i = 1, 2, \cdots. \end{cases} \tag{2.10}$$

Given α, we compute $y_0(\alpha)$, $y_1(\alpha)$, $\cdots$, $y_i(\alpha)$ until $y_{i-1}(\alpha) = y_i(\alpha)$. Computation with various α values shows that this sequence always converges. The final value of $y_i(\alpha)$ after

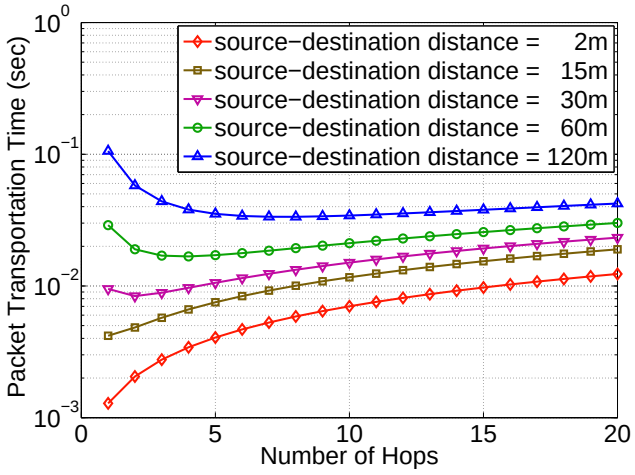


Figure 2.5: The end-to-end packet transportation time, $B = 100$ KHz, $L = 1024$ bits, $\frac{P}{N} = 10^3$, $\alpha = 2$. When $d_{v_0v_d}$=2m, R_u=2m; when $d_{v_0v_d}$=15m, 30m, 60m, R_u=15m; when $d_{v_0v_d}$=120m, R_u=17.14m.

Table 2.1: Comparison between $y(\alpha)$ and $e^\alpha - 1$

α	$y(\alpha)$	$e^\alpha - 1$	$(\frac{1}{y(\alpha)})^{\frac{1}{\alpha}}$	$(\frac{1}{e^\alpha - 1})^{\frac{1}{\alpha}}$
2	3.9216	6.3891	0.5050	0.3956
3	15.8010	19.0855	0.3985	0.3742
4	49.4353	53.5982	0.3771	0.3696
5	142.3249	147.4132	0.3710	0.3684
6	396.3833	402.4288	0.3690	0.3680

convergence is the non-zero root of Equation (2.3). Interestingly, we find $e^\alpha - 1$ is a good approximation to $y(\alpha)$. Table 2.1 compares the values of $y(\alpha)$ and $e^\alpha - 1$ for sample α values. We observe that $y(\alpha)$ is well approximated by $e^\alpha - 1$, especially for large α. Hence, $y(\alpha)$ may be obtained numerically from Equation (2.10) or approximated by Equation (2.11):

$$y(\alpha) \approx e^\alpha - 1. \tag{2.11}$$

The optimal transmission radius R is then determined by $R = (\frac{P}{Ny(\alpha)})^{\frac{1}{\alpha}}$ after $y(\alpha)$ is solved. Fig. 2.6 plots R for sample α values and signal-to-noise ratios. It is observed that R increases as the signal-to-noise ratio increases or α decreases.

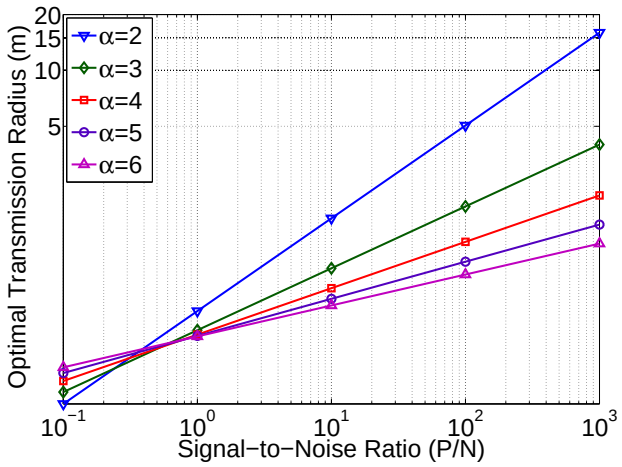


Figure 2.6: The optimal transmission radius R.

2.4 The Speed Upper Bound under Network Connectivity Constraint

We have shown that there is a unified upper bound W on $w_\varphi(t)$ for broadcast and unicast communications in large-scale networks. In this section, we study the *feasibility* of this upper bound W. Achieving W requires using the optimal transmission radius R by all the relay nodes, but R may not guarantee the network being connected. If the destination node cannot be reached by the transmission radius R, the maximum speed W is infeasible. As such, we need to understand the maximum information propagation speed constrained by the network connectivity.

2.4.1 γ-Feasible Packet Delivery

We define the term γ-*feasible delivery* to provide a probabilistic measurement on the degree of network connectivity as well as the successfulness of packet delivery. The delivery of a packet is γ-feasible if the packet can reach all the intended recipients with a probability no less than γ ($0 \leq \gamma \leq 1$). Subsequently, we define a transmission radius r to be γ-feasible (denoted as r_γ) if this r provides γ-feasible packet delivery. Note that r_γ implies that the entire network is connected with probability of at least γ by using the transmission radius r. Obviously, any transmission radius larger than r_γ is also γ-feasible. However, because R is the unique maximizer for $r \log_2(1+\frac{P}{N} r^{-\alpha})$, $w_\varphi(t)$ is maximized at the r_γ that is

closest to R, when γ-feasible packet delivery is required. We define $R_\gamma = \arg\min_{r_\gamma}\{|r_\gamma - R|\}$ to be the γ-feasible optimal transmission radius and $W_\gamma = \frac{B}{L} R_\gamma \log_2(1 + \frac{P}{N} R_\gamma^{-\alpha})$ to be the γ-feasible speed upper bound.

The network connectivity is determined by the node density and the node transmission radius. Given node density, the network is fully connected if the node transmission radius is sufficiently large. Obviously, we are able to construct a minimum spanning tree to connect every node in the network and the longest edge in this minimum spanning tree serves as a sufficiently large transmission radius. Penrose shows in [32] that the longest edge M_n in the minimum spanning tree over n Poisson distributed random nodes in a unit square satisfies

$$\lim_{n\to\infty} \Pr[n\pi M_n^2 - \log(n) \le \beta] = \exp(-e^{-\beta}).$$

Zheng [7] further proves that in an extended network with unit node density, if $\lim_{n\to\infty} c(n) = \infty$,

$$\lim_{n\to\infty} \Pr[-c(n) \le \pi M_n^2 - \log(n) \le c(n)] = 1.$$

Scaling the extended network to the dense network, we have

$$\lim_{n\to\infty} \Pr[-c(n) \le n\pi M_n^2 - \log(n) \le c(n)] = 1.$$

Choosing $c(n) = \epsilon \log(n)$ $(\epsilon > 0)$ and replacing n with λ ($n = \lambda$ in a unit square), we have

$$\lim_{\lambda\to\infty} \Pr\left[\sqrt{\frac{(1-\epsilon)\log(\lambda)}{\lambda\pi}} \le M_\lambda \le \sqrt{\frac{(1+\epsilon)\log(\lambda)}{\lambda\pi}}\right] = 1. \tag{2.12}$$

Our network consists of the tiles of unit squares with node density λ over an area $\mathcal{B}$. Thus Equation (2.12) applies to our network model too. Next, we discuss R_γ and W_γ with variable node densities in two different noise models.

2.4.2 Noise Models

We assume two noise models to determine the connectivity-constrained speed upper bound in two representative network environments. The noise models determine the scaling properties of R_γ and W_γ as node density increases. In the first one, the constant-interference model, the density of nodes scheduled for concurrent transmissions is kept constant as λ increases, i.e., $\eta\lambda$ is constant. In this case, $N = N_A + N_I$ is also a constant independent of node density. In the second one, the increasing-interference model, the percentage η of scheduled nodes is constant as λ increases such that $N = N_I$ and N is a linear function of the node density, as proven in the following lemma.

***Lemma* 2.3.** *In a network with uniform transmission power P and randomly distributed nodes in Poisson point process with density λ, the interference at any location is $N_I(\lambda) = \lambda N_I(1)$, where $N_I(1)$ is the interfence at this location when $\lambda = 1$, if the percentage η of simultaneously scheduled nodes is a constant independent of λ.*

***Remark* 2.4.** *Lemma 2.3 states the fact that the interference noise increases linearly as the node density increases.*

Proof. Since the nodes are distributed in a Poisson point process and η is independent of λ, the concurrently transmitting nodes are also in Poisson distribution with density $\eta\lambda$. By choosing two arbitrary locations $z_1, z_2 \in \mathcal{B}$ and defining N_{I,z_1,z_2} as the interference at z_1 caused by the transmissions at z_2, we have

$$\begin{aligned} N_{I,z_1,z_2}(\lambda) &= \lim_{\delta\to 0} \frac{N_{I,z_1,z_2(\delta)}(\lambda)}{\delta} \\ &= \lim_{\delta\to 0} \frac{\sum_{k=0}^{\infty} e^{-\eta\lambda\delta} \frac{(\eta\lambda\delta)^k}{k!} k P d_{z_1 z_2}^{-\alpha}}{\delta} \\ &= \eta\lambda P d_{z_1 z_2}^{-\alpha} \\ &= \lambda N_{I,z_1,z_2}(1), \end{aligned}$$

where δ is a small area around z_2 and $N_{I,z_1,z_2(\delta)}$ is the interference from δ. The total interference at z_1 from all the transmissions in the network is

$$N_{I,z_1}(\lambda) = \int_{z_2\in\mathcal{B}} N_{I,z_1,z_2}(\lambda)\,\mathrm{d}z_2 = \lambda N_{I,z_1}(1).$$

As z_1 is arbitrary, $N_I(\lambda) = \lambda N_I(1)$. □

In the increasing-interference model, since $N = N_I$, we write $N(\lambda) = \lambda N(1)$. The optimal transmission radius R in this model has the form $R = (\frac{P}{\lambda N(1) y(\alpha)})^{\frac{1}{\alpha}}$.

2.4.3 γ-Feasible Speed Upper Bound

We discover that the γ-feasible optimal transmission radius R_γ and speed upper bound W_γ can be described by the following two theorems.

***Theorem* 2.2.** *In the constant-interference noise model, there exists a threshold node density λ_C such that:*

$$R_\gamma(\lambda) = \begin{cases} R\sqrt{\frac{\lambda_C}{\lambda}} & \lambda < \lambda_C, \\ R & \lambda > \lambda_C, \end{cases} \tag{2.13}$$

$$W_\gamma(\lambda) = \begin{cases} \frac{B}{L}\big(R\sqrt{\frac{\lambda_C}{\lambda}}\big)\log_2\big(1+\frac{P}{N}\big(R\sqrt{\frac{\lambda_C}{\lambda}}\big)^{-\alpha}\big) & \lambda < \lambda_C, \\ \frac{B}{L}R\log_2(1+y(\alpha)) & \lambda > \lambda_C, \end{cases} \tag{2.14}$$

where $R = (\frac{P}{Ny(\alpha)})^{\frac{1}{\alpha}}$.

***Remark* 2.5.** *Theorem 2.2 states the fact that in the constant-interference model there is a threshold node density above which* $R_\gamma(\lambda)$ *and* $W_\gamma(\lambda)$ *are constants.*

Proof. By Equation (2.12), $\exists\ \lambda_C^{(1)}$ s.t. $\forall\ \lambda \geq \lambda_C^{(1)}$,

$$\Pr\Big[M_\lambda \leq \sqrt{\frac{(1+\epsilon)\log(\lambda)}{\lambda\pi}}\Big] \geq \gamma.$$

Let $\lambda_C^{(2)}$ denote the biggest root of the equation

$$\sqrt{\frac{(1+\epsilon)\log(\lambda)}{\lambda\pi}} = \Big(\frac{P}{Ny(\alpha)}\Big)^{\frac{1}{\alpha}}.$$

If this equation has no real root, define $\lambda_C^{(2)} = 0$. Since $\lim_{\lambda\to\infty}\sqrt{\frac{(1+\epsilon)\log(\lambda)}{\lambda\pi}} = 0$, $\forall\ \lambda \geq \lambda_C^{(2)}$,

$$\sqrt{\frac{(1+\epsilon)\log(\lambda)}{\lambda\pi}} \leq \Big(\frac{P}{Ny(\alpha)}\Big)^{\frac{1}{\alpha}}.$$

By denoting $\lambda_C = \max\{\lambda_C^{(1)}, \lambda_C^{(2)}\}$, we have $\forall\ \lambda > \lambda_C$,

$$\Pr\Big[M_\lambda \leq \Big(\frac{P}{Ny(\alpha)}\Big)^{\frac{1}{\alpha}}\Big] \geq \gamma. \tag{2.15}$$

This result indicates that when $\lambda > \lambda_C$, $R = (\frac{P}{Ny(\alpha)})^{\frac{1}{\alpha}}$ is γ-feasible, i.e., $R_\gamma(\lambda) = R$. In this case, $W_\gamma(\lambda) = \frac{B}{L}R\log_2(1+\frac{P}{N}R^{-\alpha}) = \frac{B}{L}R\log_2(1+y(\alpha))$.

When $\lambda < \lambda_C$, $R = (\frac{P}{Ny(\alpha)})^{\frac{1}{\alpha}}$ is not γ-feasible, because Inequality (2.15) does not hold. Since the network connectivity is determined by the average node degree, we can increase the node transmission radius to keep the network connected when node density is low. The analysis above shows that R is the smallest γ-feasible transmission radius with node density λ_C. By solving $\lambda\pi R'^2 = \lambda_C\pi R^2$, we obtain the smallest (also the closest to R) γ-feasible transmission radius with node density λ as $R' = R\sqrt{\frac{\lambda_C}{\lambda}}$. Hence, when $\lambda < \lambda_C$, $R_\gamma(\lambda) = R\sqrt{\frac{\lambda_C}{\lambda}}$ and $W_\gamma(\lambda) = \frac{B}{L}(R\sqrt{\frac{\lambda_C}{\lambda}})\log_2\big(1+\frac{P}{N}\big(R\sqrt{\frac{\lambda_C}{\lambda}}\big)^{-\alpha}\big)$. □

***Theorem* 2.3.** *In the increasing-interference noise model, there exists a threshold node density* λ_I *such that:*

$$R_\gamma(\lambda) = \begin{cases} R_I\sqrt{\frac{\lambda_I}{\lambda}} & \lambda < \lambda_I, \\ R & \lambda > \lambda_I, \end{cases} \tag{2.16}$$

$$W_\gamma(\lambda) = \begin{cases} \frac{B}{L}\big(R_I\sqrt{\frac{\lambda_I}{\lambda}}\big)\log_2\big(1+\frac{P}{\lambda N(1)}\big(R_I\sqrt{\frac{\lambda_I}{\lambda}}\big)^{-\alpha}\big) & \lambda < \lambda_I, \\ \frac{B}{L}R\log_2(1+y(\alpha)) & \lambda > \lambda_I, \end{cases} \tag{2.17}$$

where $R_I = (\frac{P}{\lambda_I N(1) y(\alpha)})^{\frac{1}{\alpha}}$ *and* $R = (\frac{P}{\lambda N(1) y(\alpha)})^{\frac{1}{\alpha}}$.

***Remark* 2.6.** *Theorem 2.3 states the fact that in the increasing-interference noise model* $R_\gamma(\lambda)$ *and* $W_\gamma(\lambda)$ *decrease to zero as node density increases to infinity.*

Proof. Similar to Theorem 2.2, $\exists\ \lambda_I^{(1)}$, s.t. $\forall\ \lambda \geq \lambda_I^{(1)}$,

$$\Pr\Big[M_\lambda \leq \sqrt{\frac{(1+\epsilon)\log(\lambda)}{\lambda\pi}}\Big] \geq \gamma.$$

Let $\lambda_I^{(2)}$ denote the biggest root of the equation

$$\sqrt{\frac{(1+\epsilon)\log(\lambda)}{\lambda\pi}} = \Big(\frac{P}{\lambda N(1) y(\alpha)}\Big)^{\frac{1}{\alpha}}.$$

If this equation has no real root, define $\lambda_I^{(2)} = 0$. Since $\lim_{\lambda\to\infty} \sqrt{\frac{(1+\epsilon)\log(\lambda)}{\lambda\pi}} \Big/ (\frac{P}{\lambda N(1) y(\alpha)})^{\frac{1}{\alpha}} = 0$, $\forall\ \lambda \geq \lambda_I^{(2)}$,

$$\sqrt{\frac{(1+\epsilon)\log(\lambda)}{\lambda\pi}} \leq \Big(\frac{P}{\lambda N(1) y(\alpha)}\Big)^{\frac{1}{\alpha}}.$$

By denoting $\lambda_I = \max\{\lambda_I^{(1)}, \lambda_I^{(2)}\}$, we have $\forall\ \lambda > \lambda_I$,

$$\Pr\Big[M_\lambda \leq \Big(\frac{P}{\lambda N(1) y(\alpha)}\Big)^{\frac{1}{\alpha}}\Big] \geq \gamma. \tag{2.18}$$

This shows that when $\lambda > \lambda_I$, $R = (\frac{P}{\lambda N(1) y(\alpha)})^{\frac{1}{\alpha}}$ is γ-feasible. Hence, $R_\gamma(\lambda) = R$ and $W_\gamma(\lambda) = \frac{B}{L}R\log_2(1+\frac{P}{\lambda N(1)}R^{-\alpha}) = \frac{B}{L}R\log_2(1+y(\alpha))$.

When $\lambda < \lambda_I$, similar to the discussion in Theorem 2.2, $R_I = (\frac{P}{\lambda_I N(1) y(\alpha)})^{\frac{1}{\alpha}}$ is the smallest γ-feasible transmission radius with node density λ_I and $R_I\sqrt{\frac{\lambda_I}{\lambda}}$ is the smallest (thus the closest to R) γ-feasible transmission radius with node density λ. Therefore, when $\lambda < \lambda_I$, $R_\gamma(\lambda) = R_I\sqrt{\frac{\lambda_I}{\lambda}}$ and $W_\gamma(\lambda) = \frac{B}{L}\big(R_I\sqrt{\frac{\lambda_I}{\lambda}}\big)\log_2\big(1+\frac{P}{\lambda N(1)}\big(R_I\sqrt{\frac{\lambda_I}{\lambda}}\big)^{-\alpha}\big)$. □

2.4.4 Comparison of the Noise Models

Interestingly, Theorems 2.2 and 2.3 show that the γ-feasible speed upper bound $W_\gamma(\lambda)$ behaves quite differently in the two noise models.

In the constant-interference model, when $\lambda < \lambda_C$, $W_\gamma(\lambda)$ is an increasing function of λ and reaches its maximum at $\lambda = \lambda_C$. When $\lambda > \lambda_C$, $W_\gamma(\lambda)$ is a constant, implying that

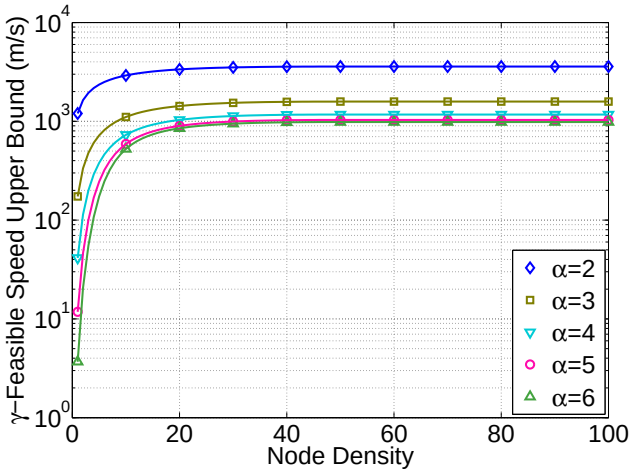


Figure 2.7: $W_\gamma(\lambda)$ in the constant-interference model, $B = 100$ KHz, $L = 1024$ bits, $\frac{P}{N} = 10^3$, $\lambda_C = 50$.

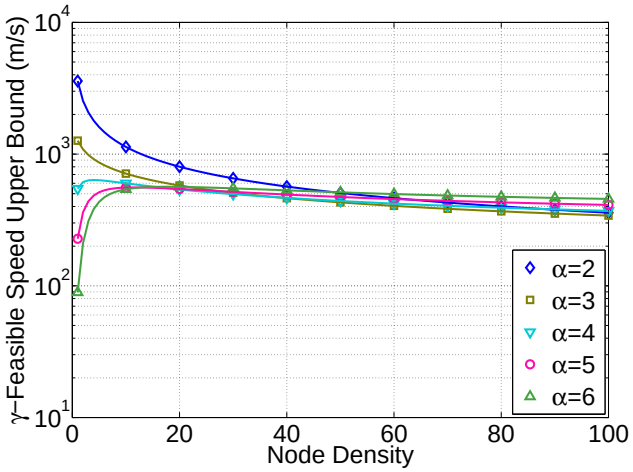


Figure 2.8: $W_\gamma(\lambda)$ in the increasing-interference model, $B = 100$ KHz, $L = 1024$ bits, $\frac{P}{N(1)} = 10^3$, $\lambda_I = 50$.

given sufficiently large node density the information propagation speed is upper bounded by a constant. An example of $W_\gamma(\lambda)$ in this model is shown in Fig. 2.7.

In the increasing-interference noise model, when $\lambda < \lambda_I$, the maximizer of $W_\gamma(\lambda)$ is obtained as $\lambda = \lambda_I \left(\frac{y(\alpha-2)}{y(\alpha)}\right)^{\frac{1}{\frac{\alpha}{2}-1}}$ by solving $\frac{\mathrm{d}W_\gamma(\lambda)}{\mathrm{d}\lambda} = 0$, where $y(\alpha-2)$ is the non-zero root of the equation

$$(1+y)\log_2(1+y) = \frac{\alpha-2}{\ln 2} y. \tag{2.19}$$

If $2 \leq \alpha \leq 3$, $y(\alpha-2)$ does not exist. In this case $W_\gamma(\lambda)$ is a decreasing function of λ when

$0 < \lambda < \lambda_I$. If $\alpha > 3$, $y(\alpha - 2)$ exists and $y(\alpha - 2) < y(\alpha)$, indicating $\lambda_I \big(\frac{y(\alpha-2)}{y(\alpha)}\big)^{\frac{1}{\frac{\alpha}{2}-1}} < \lambda_I$. Therefore, $W_\gamma(\lambda)$ increases until $\lambda = \lambda_I \big(\frac{y(\alpha-2)}{y(\alpha)}\big)^{\frac{1}{\frac{\alpha}{2}-1}}$ and then decreases thereafter. When $\lambda > \lambda_I$, $W_\gamma(\lambda)$ is always a decreasing function of λ and converges to zero as λ approaches infinity. Thus, in the increasing-interference noise model, information propagation becomes impossible when node density becomes extremely large. The strong interference prevents the transmission of any packet. An example of $W_\gamma(\lambda)$ in this model is shown in Fig. 2.8, in which it is observed that $W_\gamma(\lambda)$ takes similar values and converges in similar trend when λ is large, regardless of α.

2.5 The Gap between Achieved Speed and Upper Bound

We have shown that, given the parameter γ, there exists an optimal transmission radius $R_\gamma(\lambda)$ that may achieve the maximum information propagation speed $W_\gamma(\lambda)$ in a network with node density λ. However, as we have discussed earlier, actually achieving this maximum speed requires an additional condition that all the relay nodes are aligned and separated from each other by the distance $R_\gamma(\lambda)$. Since the nodes are distributed randomly, it may not be possible to find these perfectly located relay nodes when $\lambda < \infty$. There is always a gap between the actually achievable speed $w_\varphi(t)$ and the bound $W_\gamma(\lambda)$. We quantify this gap in this section.

Note that the speed gap exists due to the location offset of the relay nodes from their desired locations. In addition, when node scheduling is used, it may happen that the nodes closest to the desired locations are not scheduled for transmissions immediately after they receive the data packet, which increases the speed gap further. However, since we study the fastest achievable speed, we assume in this section a smart scheduling scheme that always schedules the nodes in the best locations to relay a data packet immediately after their packet reception. By using such a smart scheduling scheme, the fastest achievable speed is determined solely by the node density λ and independent of the percentage η of scheduled nodes as long as the best relay nodes are included into the schedule.

By definition, the actual information propagation speed is measured by $w_\varphi(t) = \frac{|\mathcal{L}_\varphi(t)|}{t}$. Due to the randomness of node locations, this speed may be faster or slower when the packet travels through different subareas in the network. To evaluate $w_\varphi(t)$ without

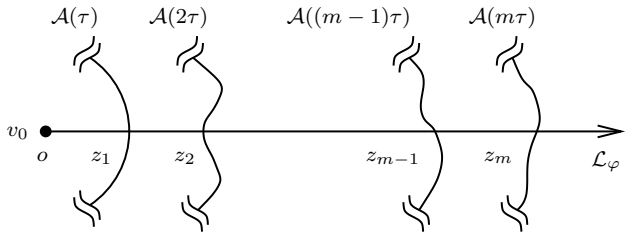


Figure 2.9: Information propagation in multihops in direction φ.

introducing the subarea bias, we define the *long-term speed* in the direction φ to be

$$w_\varphi = \lim_{t\to\infty} w_\varphi(t) = \lim_{t\to\infty} \frac{|\mathcal{L}_\varphi(t)|}{t}. \tag{2.20}$$

Since every node uses the same optimal transmission radius $R_\gamma(\lambda)$, the 1-hop transmission time $\tau = \frac{L}{B\log_2(1+\frac{P}{N}R_\gamma^{-\alpha}(\lambda))}$ is the same for every node. We rewrite Equation (2.20) as

$$w_\varphi = \lim_{m\to\infty} \frac{Z_m}{m\tau} = \lim_{m\to\infty} \frac{\sum_{i=1}^m \rho_i}{m\tau} = \frac{\overline{\rho}}{\tau}, \tag{2.21}$$

where $Z_i = d_{oz_i}$, $\rho_i = Z_i - Z_{i-1}$ and $\overline{\rho} = \lim_{m\to\infty} \frac{\sum_{i=1}^m \rho_i}{m} = E[\rho_i]$, as shown in Fig. 2.9.

First, we show that the actual information propagation speed is omnidirectional in large-scale networks. In the long term, a packet is disseminated from the source to the same distance away in any direction. The frontier of propagation is in a circular shape, as specified in the following theorem.

***Theorem* 2.4.** *In a network with homogeneous node distributions,* $\forall\ \varphi_1, \varphi_2 \in [0, 2\pi)$, $w_{\varphi_1} = w_{\varphi_2} = w$.

***Remark* 2.7.** *Theorem 2.4 states the fact that a packet reaches the same distance away in any direction after sufficiently long propagation time, though it can be faster or slower temporarily in one direction than another.*

Proof. By definition, $w_\varphi = \frac{\overline{\rho}}{\tau}$. All we need to show is $\overline{\rho}_{\varphi_1} = \overline{\rho}_{\varphi_2}$. As the nodes are distributed homogeneously, the propagation distances in φ_1 and φ_2 after i hops, Z_{i,φ_1} and Z_{i,φ_2}, are two random variables with the same probability distribution. For the same reason Z_{i-1,φ_1} and Z_{i-1,φ_2} also have the same probability distribution. Since $\rho_{i,\varphi_1} = Z_{i,\varphi_1} - Z_{i-1,\varphi_1}$ and $\rho_{i,\varphi_2} = Z_{i,\varphi_2} - Z_{i-1,\varphi_2}$, ρ_{i,φ_1} and ρ_{i,φ_2} must have the same probability distribution. Therefore, $\overline{\rho}_{\varphi_1} = E[\rho_{i,\varphi_1}] = E[\rho_{i,\varphi_2}] = \overline{\rho}_{\varphi_2}$. □

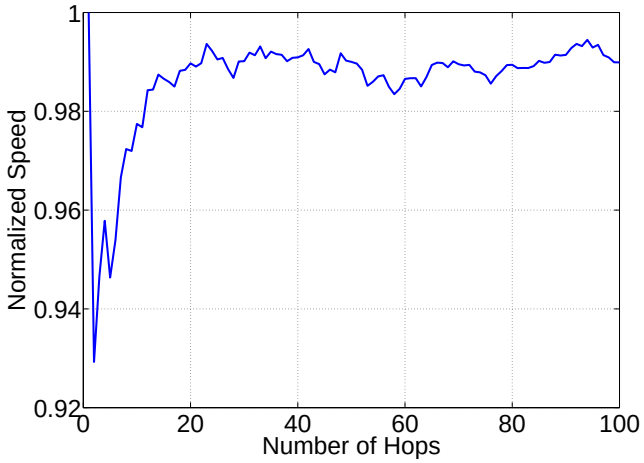


Figure 2.10: A comparison of the packet propagation speeds in six randomly chosen directions, in which the normalized speed is defined as the ratio of the minimum speed to the maximum speed in the six directions, $\lambda = 30$.

Fig. 2.10 depicts an example of the speed comparison in different directions. As the packet propagates farther away, the speeds in all directions converge to the same value.

As we will show next that w is determined by the node density λ, we write $w = w(\lambda) = \frac{\overline{\rho}(\lambda)}{\tau}$. We define the gap between the actual speed $w(\lambda)$ and its upper bound $W_\gamma(\lambda)$ as

$$\varepsilon(\lambda) = \frac{W_\gamma(\lambda) - w(\lambda)}{W_\gamma(\lambda)} = \frac{R_\gamma(\lambda) - \overline{\rho}(\lambda)}{R_\gamma(\lambda)}. \tag{2.22}$$

***Theorem* 2.5.** *In a network where the nodes are randomly distributed in a Poisson point process with density λ, $\forall\ \lambda_1 < \lambda_2$, $\varepsilon(\lambda_1) > \varepsilon(\lambda_2)$ almost surely.*

***Remark* 2.8.** *Theorem 2.5 states the fact that $\varepsilon(\lambda)$ is a strictly decreasing function of λ with probability 1.*

Proof. By definition, $\varepsilon(\lambda) = 1 - \frac{\overline{\rho}(\lambda)}{R_\gamma(\lambda)}$. To prove $\forall\ \lambda_1 < \lambda_2$, $\varepsilon(\lambda_1) > \varepsilon(\lambda_2)$, it is equivalent to show $\frac{\overline{\rho}(\lambda_1)}{R_\gamma(\lambda_1)} < \frac{\overline{\rho}(\lambda_2)}{R_\gamma(\lambda_2)}$.

First, we show $\overline{\rho}(\lambda_1) < \overline{\rho}(\lambda_2)$ almost surely. We start with a network of node density λ_2. Suppose a packet originated by node v_0 has propagated over a distance of $Z_m(\lambda_2)$ to reach location $z_m(\lambda_2)$ in an arbitrary direction φ after m hops and denote $\mathcal{P} = \{v_0, v_1, \cdots, v_{m-1}\}$ as the m-hop relay path travelled through by the packet to reach $z_m(\lambda_2)$. Now reduce the node density to λ_1 by randomly removing each node (except v_0) from the

network with probability $\frac{\lambda_2-\lambda_1}{\lambda_2}$. From the properties of Poisson process, we know that the nodes in the resulting network are Poisson distributed with density λ_1. Since removing any $v_i \in \{v_1, v_2, \cdots, v_{m-1}\}$ disrupts $\mathcal{P}$, the survival probability of $\mathcal{P}$ is

$$\Pr[\mathcal{P} \text{ survives}] = \Big(\frac{\lambda_1}{\lambda_2}\Big)^{m-1}.$$

When $m \to \infty$, $\Pr[\mathcal{P} \text{ survives}] \to 0$, implying $z_m(\lambda_2)$ is almost unreachable in the resulting network. Denoting $Z_m(\lambda_1)$ as the propagation distance of the packet in direction φ after m hops in the resulting network, we have $\lim_{m\to\infty} \Pr[Z_m(\lambda_1) < Z_m(\lambda_2)] = 1$, which gives

$$\Pr[\overline{\rho}(\lambda_1) < \overline{\rho}(\lambda_2)] = \lim_{m\to\infty} \Pr\Big[\frac{Z_m(\lambda_1)}{m} < \frac{Z_m(\lambda_2)}{m}\Big] = 1,$$

i.e., $\overline{\rho}(\lambda_1) < \overline{\rho}(\lambda_2)$ almost surely.

Next, we show $R_\gamma(\lambda_1) \geq R_\gamma(\lambda_2)$. Theorems 2.2 and 2.3 indicate that $R_\gamma(\lambda)$ is a decreasing function (not always strictly though) of λ in both noise models. Hence, $\forall$ $\lambda_1 < \lambda_2$, $R_\gamma(\lambda_1) \geq R_\gamma(\lambda_2)$.

Combining $\overline{\rho}(\lambda_1) < \overline{\rho}(\lambda_2)$ and $R_\gamma(\lambda_1) \geq R_\gamma(\lambda_2)$, we obtain $\frac{\overline{\rho}(\lambda_1)}{R_\gamma(\lambda_1)} < \frac{\overline{\rho}(\lambda_2)}{R_\gamma(\lambda_2)}$ almost surely. □

Theorem 2.5 points out that $\varepsilon(\lambda)$ reduces as λ increases. The next theorem provides a quantified measurement of $\varepsilon(\lambda)$.

***Theorem* 2.6.** *In a network where the nodes are randomly distributed in a Poisson point process with density λ and the optimal transmission radius $R_\gamma(\lambda)$ is used, defining $a = \lambda\pi R_\gamma^2(\lambda)$, $g_1(a) = \int_0^1 e^{a(x^2-1)}\mathrm{d}x$ and $g_2(a) = \int_0^1 e^{-\frac{1}{3}ax^2}\mathrm{d}x$,*

$$g_1(a) \leq \varepsilon(\lambda) \leq g_2(a). \tag{2.23}$$

***Remark* 2.9.** *Theorem 2.6 provides the bounds on the convergence rate of the speed gap as the node density increases.*

Proof. First, we define two relevant random variables that will be used in this proof. As depicted in Fig. 2.11(a), we define X_{max} as the distance from a node to its farthest neighbor within the transmission radius $R_\gamma(\lambda)$. In Fig. 2.11(b), we draw a sector at an arbitrary location o with radius $R_\gamma(\lambda)$ and central angle $\frac{2\pi}{3}$, and define X_{min} as the distance from o to the nearest node found in this sector.

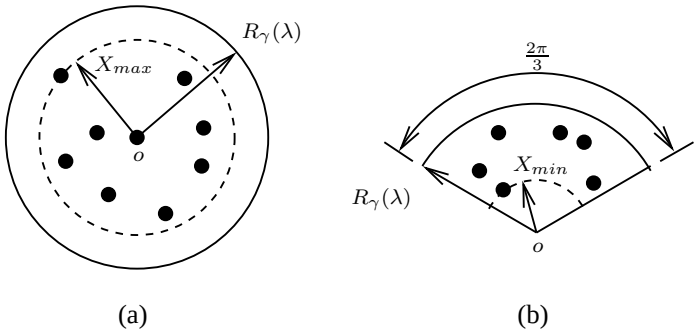


Figure 2.11: Definitions of X_{max} and X_{min}.

Next, we prove $\varepsilon(\lambda) \geq g_1(a)$. As shown in Fig. 2.12, letting $\mathcal{P} = \{v_0, v_1, \cdots, v_{m-1}\}$ denote the relay path travelled by a packet from v_0 to reach z_m in m hops,

$$Z_m \leq \sum_{i=0}^{m-2} d_{v_i v_{i+1}} + d_{v_{m-1} z_m} \leq \sum_{i=0}^{m-1} X_{max,i} + R_\gamma(\lambda),$$

where $X_{max,i}$ is the X_{max} occuring at v_i. Then,

$$\begin{aligned}
\overline{\rho}(\lambda) &= \lim_{m\to\infty} \frac{Z_m}{m} \\
&\leq \lim_{m\to\infty} \frac{\sum_{i=0}^{m-1} X_{max,i} + R_\gamma(\lambda)}{m} \\
&= \lim_{m\to\infty} \frac{\sum_{i=0}^{m-1} X_{max,i}}{m} \\
&= E[X_{max}],
\end{aligned}$$

since $X_{max,i}$ has i.i.d. probability distribution. We obtain $E[X_{max}]$ as follows. According to the Poisson distribution, with probability e^{-a}, a node v_i has no neighbor (i.e., $X_{max} = 0$). With probability $1-e^{-a}$, v_i has at least one neighbor (i.e., $X_{max} > 0$). Given $0 < x \leq R_\gamma(\lambda)$,

$$\begin{aligned}
\Pr[X_{max} \leq x | X_{max} > 0] &= \frac{1}{1-e^{-a}} \sum_{k=1}^{\infty} e^{-a} \frac{a^k}{k!} \Big(\frac{\pi x^2}{\pi R_\gamma^2(\lambda)} \Big)^k \\
&= \frac{e^{-a}}{1-e^{-a}} \Big(e^{\frac{ax^2}{R_\gamma^2(\lambda)}} - 1 \Big).
\end{aligned}$$

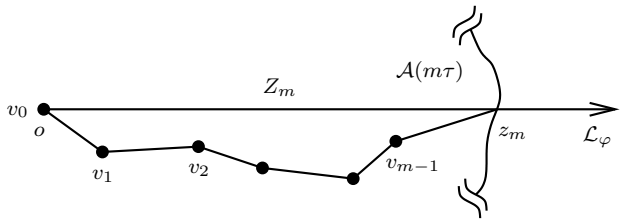


Figure 2.12: The propagation distance Z_m in direction φ.

The conditional expectation is

$$\begin{aligned} E[X_{max}|X_{max}>0] &= \int_0^{R_\gamma(\lambda)} x \ \mathrm{d}\Pr[X_{max}\le x|X_{max}>0] \\ &= \int_0^{R_\gamma(\lambda)} \frac{e^{-a}}{1-e^{-a}}\Big(\frac{2ax^2}{R_\gamma^2(\lambda)}\Big)e^{\frac{ax^2}{R_\gamma^2(\lambda)}}\mathrm{d}x \\ &= \frac{R_\gamma(\lambda)}{1-e^{-a}}\Big(1-\int_0^1 e^{a(x^2-1)}\mathrm{d}x\Big). \end{aligned}$$

The unconditional expectation is

$$\begin{aligned} E[X_{max}] &= e^{-a}\cdot 0+(1-e^{-a})\cdot E[X_{max}|X_{max}>0] \\ &= R_\gamma(\lambda)\Big(1-\int_0^1 e^{a(x^2-1)}\mathrm{d}x\Big). \end{aligned}$$

Thus,

$$\varepsilon(\lambda)\ge\frac{R_\gamma(\lambda)-E[X_{max}]}{R_\gamma(\lambda)}=\int_0^1 e^{a(x^2-1)}\mathrm{d}x.$$

Finally, we prove $\varepsilon(\lambda)\le g_2(a)$. As Fig. 2.13 illustrates, denote z_{m-1} as the farthest location that a packet has reached in direction φ after $m-1$ hops and $\mathcal{P}=\{v_0,v_1,\cdots,v_{m-2}\}$ as the relay path travelled by the packet to reach z_{m-1}. Draw a sector at z_{m-1} with radius $R_\gamma(\lambda)$ and central angle $\frac{2\pi}{3}$, as illustrated by the dash-line encompassed area in Fig. 2.13, where $\angle u_4z_{m-1}u_6=\angle u_5z_{m-1}u_6=\frac{\pi}{3}$. Note that for any node v' in this sector, $d_{v_{m-2}v'}\le R_\gamma(\lambda)$, implying that v' must have received the packet by time $(m-1)\tau$ and forwarded the packet by time $m\tau$, i.e., $v'\in\widetilde{\mathcal{V}}(m\tau)$. Since z_m is the farthest location from o on $\mathcal{L}_\varphi$ covered by $\widetilde{\mathcal{V}}(m\tau)$, $d_{oz_m}\ge d_{ou_2}$, where u_2 is the farthest location reached by v' on $\mathcal{L}_\varphi$. So, $\rho_m=d_{z_{m-1}z_m}\ge d_{z_{m-1}u_2}$. By triangle inequality,

$$\rho_m\ge d_{z_{m-1}u_2}\ge d_{u_1u_2}-d_{u_1z_{m-1}}=R_\gamma(\lambda)-d_{u_1z_{m-1}}.$$

As the above inequality holds for all the v' in the sector,

$$\rho_m \geq \max_{\{v'\}}\{R_\gamma(\lambda) - d_{u_1 z_{m-1}}\} = R_\gamma(\lambda) - X_{min},$$

where X_{min} is defined in Fig. 2.11(b). Replacing m with i,

$$\begin{aligned}
\overline{\rho}(\lambda) &= \lim_{m\to\infty} \frac{\sum_{i=1}^{m} \rho_i}{m} \\
&\geq R_\gamma(\lambda) - \lim_{m\to\infty} \frac{\sum_{i=1}^{m} X_{min,i}}{m} \\
&= R_\gamma(\lambda) - E[X_{min}]
\end{aligned}$$

where $X_{min,i}$ is the X_{min} occuring at z_{i-1} and $X_{min,i}$ has i.i.d. probability distribution. Next, we compute $E[X_{min}]$. We know from the Poisson distribution that with probability $e^{-\frac{1}{3}a}$ there is no node in the sector (i.e., $X_{min} = R_\gamma(\lambda)$) and with probability $1-e^{-\frac{1}{3}a}$ there is at least one node in the sector (i.e., $X_{min} < R_\gamma(\lambda)$). Given $0 \leq x < R_\gamma(\lambda)$,

$$\begin{aligned}
\Pr[X_{min} \leq x | X_{min} < R_\gamma(\lambda)] &= 1 - \Pr[X_{min} > x | X_{min} < R_\gamma(\lambda)] \\
&= 1 - \frac{e^{-\frac{1}{3}\lambda\pi x^2} - e^{-\frac{1}{3}a}}{1 - e^{-\frac{1}{3}a}}.
\end{aligned}$$

The conditional expectation is

$$\begin{aligned}
E[X_{min} | X_{min} < R_\gamma(\lambda)] &= \int_0^{R_\gamma(\lambda)} x \, \mathrm{d}\Pr[X_{min} \leq x | X_{min} < R_\gamma(\lambda)] \\
&= \int_0^{R_\gamma(\lambda)} \frac{\frac{2}{3}\lambda\pi x^2 e^{-\frac{1}{3}\lambda\pi x^2}}{1 - e^{-\frac{1}{3}a}} \\
&= \frac{R_\gamma(\lambda)}{1 - e^{-\frac{1}{3}a}} \Big(\int_0^1 e^{-\frac{1}{3}ax^2} \mathrm{d}x - e^{-\frac{1}{3}a} \Big).
\end{aligned}$$

The unconditional expectation is

$$\begin{aligned}
E[X_{min}] &= e^{-\frac{1}{3}a} R_\gamma(\lambda) + (1 - e^{-\frac{1}{3}a}) E[X_{min} | X_{min} < R_\gamma(\lambda)] \\
&= R_\gamma(\lambda) \int_0^1 e^{-\frac{1}{3}ax^2} \mathrm{d}x.
\end{aligned}$$

Thus,

$$\varepsilon(\lambda) \leq \frac{R_\gamma(\lambda) - (R_\gamma(\lambda) - E[X_{min}])}{R_\gamma(\lambda)} = \int_0^1 e^{-\frac{1}{3}ax^2} \mathrm{d}x.$$

□

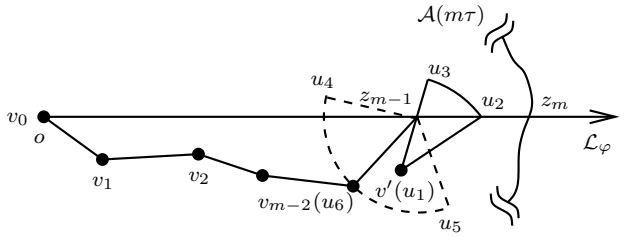
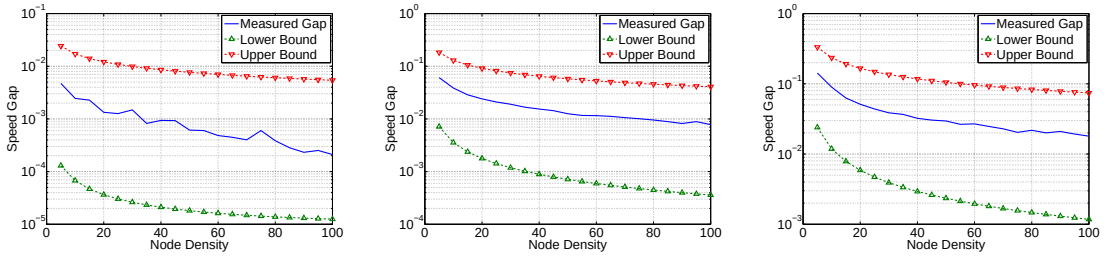
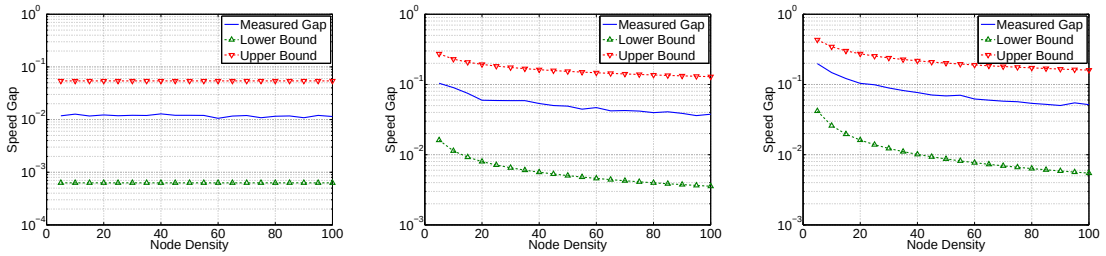


Figure 2.13: The mth-hop propagation distance $\rho_m = Z_m - Z_{m-1}$ in direction φ, $d_{z_{m-1}u_4} = d_{z_{m-1}u_5} = d_{z_{m-1}u_6} = d_{u_1u_2} = d_{u_1u_3} = R_\gamma(\lambda)$, $\angle u_4 z_{m-1} u_6 = \angle u_5 z_{m-1} u_6 = \frac{\pi}{3}$.



(a) $\alpha = 2$ (b) $\alpha = 4$ (c) $\alpha = 6$

Figure 2.14: The speed gap in the constant-interference noise model, $\frac{P}{N} = 10^3$.



(a) $\alpha = 2$ (b) $\alpha = 4$ (c) $\alpha = 6$

Figure 2.15: The speed gap in the increasing-interference noise model, $\frac{P}{N(1)} = 10^3$.

The speed gap $\varepsilon(\lambda)$ and its bounds are shown in Fig. 2.14 and 2.15 for different α values. Based on the result of Theorem 2.6, we are able to determine the asymptotic convergence rate of the speed gap as the node density approaches infinity. In order to present this asymptotic rate, we first introduce Lemma 2.4.

***Lemma* 2.4.** *Define* $h_1(b) = \int_0^1 k^{b(x^2-1)} \mathrm{d}x$ *and* $h_2(b) = \int_0^1 k^{-\frac{1}{3}bx^2} \mathrm{d}x$, *where* $k > 1$, $b > 0$. $\forall\ 0 < c < 1$ *and* $\epsilon > 0$, $h_1(b) > c^b$ *and* $h_2(b) < c^{b^{1-\epsilon}}$ *as* $b \to \infty$.

***Remark* 2.10.** *Lemma 2.4 states the fact that the tails of $h_1(b)$ and $h_2(b)$ are bounded by exponential functions.*

Proof. When $0 < c < 1$, $\exists\ 0 < x_0 < 1$ s.t. $k^{x_0^2-1} > c$. Because $\forall\ b > 0$,

$$\int_0^1 \Big(\frac{k^{x^2-1}}{c}\Big)^b \mathrm{d}x \geq \int_{x_0}^1 \Big(\frac{k^{x^2-1}}{c}\Big)^b \mathrm{d}x \geq (1-x_0)\Big(\frac{k^{x_0^2-1}}{c}\Big)^b,$$

as $b \to \infty$,

$$\int_0^1 \Big(\frac{k^{x^2-1}}{c}\Big)^b \mathrm{d}x \geq (1-x_0)\Big(\frac{k^{x_0^2-1}}{c}\Big)^b \to \infty > 1.$$

Therefore, $h_1(b) = \int_0^1 k^{b(x^2-1)}\mathrm{d}x > c^b$ as $b \to \infty$.

When $0 < c < 1$, $\epsilon > 0$ and $x > 0$, as $b \to \infty$, $c^{b^{-\epsilon}} \to 1$ and $\big(\frac{k^{-\frac{1}{3}x^2}}{c^{b^{-\epsilon}}}\big)^b \to 0$. Hence, $\int_0^1 \big(\frac{k^{-\frac{1}{3}x^2}}{c^{b^{-\epsilon}}}\big)^b \mathrm{d}x \to 0 < 1$, which gives $h_2(b) = \int_0^1 k^{-\frac{1}{3}bx^2}\mathrm{d}x < c^{b^{1-\epsilon}}$ as $b \to \infty$. □

The asymptotic convergence rate of $\varepsilon(\lambda)$ is then summarized in the next two theorems.

***Theorem* 2.7.** *In the constant-interference noise model, $\exists\ 0 < c_1, c_2 < 1$ and $\forall\ 0 < c < 1$, $\epsilon > 0$,*

$$\begin{cases} c_1 \leq \varepsilon(\lambda) \leq c_2 & \lambda < \lambda_C, \\ c^\lambda < \varepsilon(\lambda) < c^{\lambda^{1-\epsilon}} & \lambda > \lambda_C, \text{ as } \lambda \to \infty. \end{cases}$$

***Remark* 2.11.** *Theorem 2.7 states the fact that in the constant-interference noise model the speed gap converges to zero exponentially with exponent $\lambda^{1-\epsilon}$, where ϵ is an arbitrarily small positive real number.*

Proof. When $\lambda < \lambda_C$, $R_\gamma(\lambda) = R\sqrt{\frac{\lambda_C}{\lambda}}$ and $a = \lambda_C \pi R^2$. By Theorem 2.6, letting $c_1 = g_1(\lambda_C \pi R^2)$ and $c_2 = g_2(\lambda_C \pi R^2)$, we have $c_1 \leq \varepsilon(\lambda) \leq c_2$.

When $\lambda > \lambda_C$, $R_\gamma(\lambda) = R$ and $a = \lambda \pi R^2$. Choose $k = e^{\pi R^2}$ and $b = \lambda$. By Theorem 2.6 and Lemma 2.4, $\forall\ 0 < c < 1$ and $\epsilon > 0$, as $\lambda \to \infty$,

$$\varepsilon(\lambda) \geq g_1(a) = h_1(\lambda) > c^\lambda,$$

and

$$\varepsilon(\lambda) \leq g_2(a) = h_2(\lambda) < c^{\lambda^{1-\epsilon}}.$$

□

***Theorem* 2.8.** *In the increasing-interference noise model,* $\exists\ 0 < c_1, c_2 < 1$ *and* $\forall\ 0 < c < 1$, $\epsilon > 0$,

$$\begin{cases} c_1 \leq \varepsilon(\lambda) \leq c_2 & \lambda < \lambda_I, \\ c^{\lambda^{1-\frac{2}{\alpha}}} < \varepsilon(\lambda) < c^{\lambda^{(1-\frac{2}{\alpha})(1-\epsilon)}} & \lambda > \lambda_I, \text{ as } \lambda \to \infty. \end{cases}$$

***Remark* 2.12.** *Theorem 2.8 states the fact that in the increasing-interference noise model the speed gap converges to zero exponentially with exponent* $\lambda^{(1-\frac{2}{\alpha})(1-\epsilon)}$*, where* ϵ *is an arbitrarily small positive real number.*

Proof. When $\lambda < \lambda_I$, $R_\gamma(\lambda) = R_I\sqrt{\frac{\lambda_I}{\lambda}}$ and $a = \lambda_I \pi R_I^2$. By Theorem 2.6, letting $c_1 = g_1(\lambda_I \pi R_I^2)$ and $c_2 = g_2(\lambda_I \pi R_I^2)$, we have $c_1 \leq \varepsilon(\lambda) \leq c_2$.

When $\lambda > \lambda_I$, $R_\gamma(\lambda) = \left(\frac{P}{\lambda N(1) y(\alpha)}\right)^{\frac{1}{\alpha}}$ and $a = \lambda^{1-\frac{2}{\alpha}} \pi (\frac{P}{N(1)y(\alpha)})^{\frac{2}{\alpha}}$. Choose $k = e^{\pi(\frac{P}{N(1)y(\alpha)})^{\frac{2}{\alpha}}}$ and $b = \lambda^{1-\frac{2}{\alpha}}$. By Theorem 2.6 and Lemma 2.4, $\forall\ 0 < c < 1$ and $\epsilon > 0$, as $\lambda \to \infty$,

$$\varepsilon(\lambda) \geq g_1(a) = h_1(\lambda^{1-\frac{2}{\alpha}}) > c^{\lambda^{1-\frac{2}{\alpha}}},$$

and

$$\varepsilon(\lambda) \leq g_2(a) = h_2(\lambda^{1-\frac{2}{\alpha}}) < c^{\lambda^{(1-\frac{2}{\alpha})(1-\epsilon)}}.$$

□

Theorems 2.7 and 2.8 reveal that in both noise models there is a threshold node density, below which $\varepsilon(\lambda)$ is bounded by constants (the constants are determined by the choice of parameter γ) and above which $\varepsilon(\lambda)$ converges to zero exponentially in the rates of $c^{\lambda^{1-\epsilon}}$ and $c^{\lambda^{(1-\frac{2}{\alpha})(1-\epsilon)}}$ respectively.

2.6 The Fastest Packet Transmission Algorithm

Based on the analysis, we know that a packet is transmitted at the fastest speed if the next-hop node is located at a distance of R from the forwarding node and in the direction towards the destination. In this section, we design a routing algorithm that identifies the next-hop relay nodes to achieve the fastest packet transmission. As this new algorithm assumes the knowledge of node locations, it is a variant of the geographic routing algorithms. Specifically, we assume the following information is available to every

The Fastest Packet Transmission Algorithm

1 determine $y(\alpha)$ by Eq. (2.10) or (2.11) and $R = (\frac{P}{Ny(\alpha)})^{\frac{1}{\alpha}}$
2 **if** $\frac{R}{d_{v_i v_d}} \geq 1$
3 **then** compute $C = B\log_2(1 + \frac{P}{N} d_{v_i v_d}^{-\alpha})$
4 transmit the packet to v_d at rate C
5 **else** determine $\mathbf{z} = (1 - \frac{R}{d_{v_i v_d}})\mathbf{v_i} + \frac{R}{d_{v_i v_d}}\mathbf{v_d}$
6 find node $v_{i+1} = \arg_v \min\{d_{zv}\}$
7 compute $C = B\log_2(1 + \frac{P}{N} d_{v_i v_{i+1}}^{-\alpha})$
8 transmit the packet to v_{i+1} at rate C

Figure 2.16: The fastest packet transmission algorithm running on node v_i that forwards a packet to destination v_d. The vectors $\mathbf{v_i}$ and $\mathbf{v_d}$ denote the locations of v_i and v_d. The environmental parameters α, B, P and N are constant and known.

node [33]: 1) the location of the node itself, 2) the location of the packet destination, and 3) the locations of the neighbor nodes.

By assuming the availability of location information, we design a fastest packet transmission algorithm as in Fig. 2.16, which runs on every node in the network distributively and independently. To identify the next-hop node, a forwarding node v_i finds out the transmission radius R first. $R = (\frac{P}{Ny(\alpha)})^{\frac{1}{\alpha}}$ can be either accurately determined by Eq. (2.10) or approximated by Eq. (2.11). Next, if $R \geq d_{v_i v_d}$ where v_d is the destination node, the packet is sent directly to v_d. In this case, v_i computes the capacity C of the wireless link connecting itself and the destination node v_d, and then sends the packet at the computed bit rate C. Higher rate than C will cause incorrect reception at v_d and lower rate than C will introduce extra unnecessary transmission delay. Otherwise, if $R < d_{v_i v_d}$, at least one more relay node is needed. In this case, v_i first determines the desired location of the next-hop node, then finds the node closest to the desired location, and finally computes the link capacity to send the packet at the correct rate to the identified next-hop node.

2.7 Summary

In this chapter we have studied the packet delay problem in lightly-loaded large-scale multi-hop wireless networks in terms of the packet propagation speed. We find that there exists an upper bound, determined by the network parameters, on the information propagation speed. This upper bound is different for broadcast communications and unicast communications, but the two bounds converge in large-scale networks. As a necessary

condition for achieving this upper bound, all the relay nodes must use an optimal transmission radius. We also reveal that, when network connectivity is considered, the feasible speed upper bound is a function of node density. If the noise in the network is constant, the speed bound is constant when node density exceeds a threshold. Otherwise, if the noise is an increasing function of node density, the speed bound decreases to zero as node density grows to infinity. Finally, we prove that a packet propagates omnidirectionally in large-scale random networks and the gap between its actual speed and the upper bound decreases exponentially when node density increases to infinity. The work in this chapter provides fundamental understanding of the achievable fastest information delivery in large-scale wireless networks, which is instrumental to the delay-minimization routing protocol design in wireless networks. The speed upper bound found in this chapter also applies to the wireless networks with arbitrary traffic loads and routing protocols, since heavy load and non-optimal path selection incur extra packet transportation delay. The tightness of the speed upper bound, however, will need to be reconsidered in such cases.

Chapter 3

The Capacity of Wireless Networks

Our study in the previous chapter has determined the fastest information propagation in large wireless networks. In this chapter, we will investigate another important performance aspect, the network capacity. Specifically, we want to determine the scaling of network capacity as the network size increases. The existing results in the literature have been obtained based on the implicit assumption of negligible overhead in acquiring the network topology and synchronizing the link transmissions. In large networks, however, global topology collection and global link synchronization are infeasible with both the centralized and the distributed link scheduling schemes. This gap between the well-known capacity results and the impractical assumption on link scheduling potentially undermines our understanding of the *achievable* network capacity. Therefore, the following question remains open: can *localized* scheduling algorithms achieve the same order of capacity as their global counterpart? In this chapter, we propose the *scheduling partition* methodology by decomposing a large network into many small autonomous scheduling zones, in which localized scheduling algorithms are implemented independently from one another. We prove that any localized scheduling algorithm that satisfies a set of sufficient and necessary conditions yields the same order of capacity as the widely assumed global scheduling strategy. In comparison to the network dimension $\sqrt{n}$, scheduling partition sizes $\Theta(\sqrt{\log n})$ and $\Theta(1)$ are sufficient for optimal capacity scaling with random and arbitrary node placements respectively. We finally propose a distributed partition protocol and a localized scheduling algorithm that achieve the order optimal capacity in large wireless networks. Our results provide a practical solution for maximizing the capacity of large-scale wireless networks.

3.1 Motivation and Related Work

Since the seminal work [1], many efforts have been made to find the upper and lower capacity bounds for different communication types in various network settings. On one hand, the constraints of wireless interference and multihop relay prevent a large wireless network from reaching arbitrarily high throughput, while on the other hand, thoughtful link scheduling can avoid transmission collisions such that the achievable throughput is within a constant fraction of the upper bound. The work in [1] considered the throughput of unicast communications. The study was extended to broadcast communications in [7, 35, 36] and later to multicast communications in [37, 38]. Recently, the capacity results of unicast, broadcast and multicast were unified by introducing the concept of (n, m, k)-casting [39]. Another important discovery on wireless network capacity was made by utilizing the percolation method [40], which improves the multicast capacity bounds when nodes are randomly placed in a network [41]. All the existing results were however obtained by assuming a globally collision-free link scheduling scheme, which may not be available in large wireless networks.

Collision-free transmissions are implemented in wireless networks via link scheduling schemes. The goal of scheduling is to maximize the network throughput while ensuring every transmission to be successful, i.e., to find the shortest schedule that fulfills all the link transmission requests. In the approach of centralized scheduling [42–46], a designated entity collects the entire network topology information and computes a schedule that requires the interfering links to transmit in different time slots. As network size increases, centralized scheduling clearly becomes infeasible because of the overhead for global topology collection. The distributed scheduling algorithms [47–51] work better in large networks by not collecting global topology. Instead, nodes exchange messages with neighbors to reach a consensus regarding their respective transmission times. As a result of the consensus, a maximal set of collision-free links, called *maximal matching* [52], transmit simultaneously while the interfering links transmit only at different times. However, distributed scheduling is not a feasible solution for large wireless networks either. The amount of exchanged messages and the latency to determine the transmission schedule increase as network size grows. Furthermore, network-wide clock synchronization is required in order for each link to adhere to the established schedule. Any clock error may result in the failure of correct timing in link transmissions. Besides scheduling, random access [53, 54] has also been studied in the

literature as an alternative approach to link coordination, but the capacity bounds with random access are unknown yet in large wireless networks. With an increasing network size, the competition among neighboring nodes for medium access may become intense and result in high chance of transmission failures.

Therefore, the inapplicability of existing link scheduling schemes in large wireless networks is in sharp contrast to the assumption of a globally collision-free link transmission schedule in the current study of network capacity scaling. The gap between them presents an open question: can localized link scheduling algorithms achieve similar capacity scaling as their global counterpart? The answer to this question is important as it bridges the disconnection between the theoretical capacity results and the practical implementation toward capacity maximization.

In this chapter, we propose a *scheduling partition* methodology to address the feasibility of maximum capacity scaling in large-scale wireless networks. We decompose a large wireless network into many small partitions with the links in each partition scheduled independently from other partitions. The network decomposition approach thus significantly reduces the scheduling complexity as compared to the existing algorithms. When designed properly, the scheduling complexity is constant for each partition, regardless of the network size. Note that scheduling partition may introduce link collisions due to the absence of coordination across partitions, which breaches the requirement for collision-free scheduling. Nevertheless, we provide a set of partition and scheduling principles that guarantee the infringement not to jeopardize the objective of maximum capacity scaling. Guided by our results on scheduling partition, we propose a distributed protocol for network decomposition and a localized algorithm for link scheduling in each partition. Our method achieves the order optimal capacity in large wireless networks with constant scheduling complexity regardless of network size.

3.2 Problem Formulation

A variety of network models have been used in the literature to represent different scenarios of network expansions, node communications, location distributions and wireless interferences. We consider all these models to study a generally applicable link scheduling method in wireless networks.

3.2.1 Network Models

We consider the network expansion as the well-known *extended network* model, which was initially introduced in [1] and later widely used for wireless network capacity study, e.g., [7, 12, 34–41]. The extended network is characterized by n nodes distributed in a square region $\mathcal{B}$ with area $|\mathcal{B}| = n$. As network scales $n \to \infty$, the node density keeps constant 1. Another popular scaling model, the *dense network*, differs from the extended model by a geographical factor $\sqrt{n}$. Although we will proceed with the extended model in the next, our results apply to both the extended and the dense networks. In addition to network sizes, we also consider the following models for wireless interferences, node locations and communication scenarios.

Interference Models

Three models are widely used in the literature to represent wireless interference: the *protocol* model, the *physical* model and the *generalized physical* model.

The protocol model ($\mathbb{I}_{\text{prot}}$) [1] specifies a successful transmission from node v_i to node v_j if

$$|X_i - X_j| \leq r(n), \tag{3.1}$$

and for any other simultaneously transmitting node v_k

$$|X_k - X_j| \geq (1 + \Delta)r(n), \tag{3.2}$$

where X_i, X_j and X_k are the locations of v_i, v_j and v_k, $r(n)$ is the critical transmission radius of all nodes, and Δ models the guard zone around v_j in which any simultaneous transmission causes collision at v_j. Whenever a transmission is successful, v_i communicates to v_j at a constant data rate $w_{ij} = W$. Otherwise, whenever collision occurs, $w_{ij} = 0$.

The physical model ($\mathbb{I}_{\text{phy}}$) [1] requires a minimum Signal-to-Interference-plus-Noise-Ratio (SINR) at v_j in order for a transmission from v_i to v_j to be successful, as shown below.

$$\frac{\frac{P_i}{|X_i - X_j|^\alpha}}{BN_0 + \sum_{k \neq i} \frac{P_k}{|X_k - X_j|^\alpha}} \geq \beta, \tag{3.3}$$

where v_k is any node that transmits simultaneously, $P_{\min} \leq P_i, P_k \leq P_{\max}$ are the transmission powers, B is the spectrum bandwidth, N_0 is the spectrum density of ambient noise,

$\alpha > 2$ is the path loss exponent, and β is a constant. We assume that all the transmissions occur within radius $r(n)$. In order to overcome the singularity problem [55] that occurs when v_i and v_j are arbitrarily close and the received signal power at v_j is amplified unrealistically by the path loss model, we assume a minimum distance $r_0(n) = \varepsilon r(n)$ $(0 < \varepsilon < 1)$ for every transmission. In summary, $r_0(n) \leq |X_i - X_j| \leq r(n)$. Besides, we assume the ambient noise is non-negligible as compared to the received signal power, i.e., $\gamma_1 BN_0 \leq P_{\min} r^{-\alpha}(n) \leq P_{\max} r^{-\alpha}(n) \leq \gamma_2 BN_0$ where $\gamma_1, \gamma_2 > 0$ are constants. The data rate is $w_{ij} = W$ for successful transmissions or $w_{ij} = 0$ for failed transmisssions.

The generalized physical model ($\mathbb{I}_{\text{gen}}$) [36] differs from the physical model in the data rate w_{ij}, which is determined as

$$w_{ij} = B \log_2 \Big(1 + \frac{\frac{P_i}{|X_i - X_j|^\alpha}}{BN_0 + \sum_{k \neq i} \frac{P_k}{|X_k - X_j|^\alpha}}\Big). \tag{3.4}$$

As in the physical model, we assume $P_{\min} \leq P_i, P_k \leq P_{\max}$, $\alpha > 2$, $r_0(n) \leq |X_i - X_j| \leq r(n)$, and $\gamma_1 BN_0 \leq P_{\min} r^{-\alpha}(n) \leq P_{\max} r^{-\alpha}(n) \leq \gamma_2 BN_0$. Note that due to non-negligence of BN_0, the variable link data rate is upper bounded by

$$w_{ij} \leq B \log_2(1 + \frac{P_{\max}(\varepsilon r(n))^{-\alpha}}{BN_0}) \leq B \log_2(1 + \gamma_2 \varepsilon^{-\alpha}). \tag{3.5}$$

Location Models

We consider two prevailing node location models: *random* and *arbitrary*. In random networks ($\mathbb{L}_{\text{rand}}$) [1], node locations are distributed in a random Poisson point process. As $n \to \infty$, $r(n) = \Theta(\sqrt{\log n})$ is required for network connectivity. In arbitrary networks ($\mathbb{L}_{\text{arbi}}$) [1], node locations are assigned in need. One example is to place the nodes on a grid with equal distance between neighbors. In this case, as $n \to \infty$, $r(n) = \Theta(1)$ is sufficient for network connectivity.

Communication Models

Three communication models are studied in the literature: *unicast*, *broadcast* and *multicast*. The unicast model ($\mathbb{C}_{\text{uni}}$) [1, 12, 34] assumes n source-destination pairs. Every node in the network is a source and it selects another node randomly as its destination. In the broadcast model ($\mathbb{C}_{\text{bro}}$) [7,35], each node disseminates its packets to all the other $n-1$ nodes. As the transition between unicast and broadcast, the multicast model ($\mathbb{C}_{\text{mul}}$) [36–38]

disseminates the packets from each node to $k-1$ randomly chosen destinations. It is equivalent to unicast if $k=2$ and to broadcast if $k=n$.

From a single transmission point of view, these communication models differ in the number of receivers during one transmission. In unicast there is only one receiver for each transmission, while in broad/multi-cast there could be multiple receivers at the branching points of the broad/multi-cast tree. For presentation convenience, we define the group of links sharing the same branching point in a broad/multi-cast tree as *broad/multi-cast branches* when the branching node sends the same packet along these links. In addition, we define the number of communication sessions that traverse a node v_i as the *node session degree* ζ_i and the number of sessions that traverse a link l_{ij} (l_{ij} is directional) as the *link session degree* ζ_{ij}. We further define $\eta_{ij}=\frac{\zeta_{ij}}{\zeta_i}$ to denote the percentage of time spent on l_{ij} when v_i is transmitting. Finally, we assume the packet length L bounded by $L_{\min}\leq L\leq L_{\max}$.

3.2.2 Capacity Formulation

In this chapter, we propose a scheduling partition methodology for achieving the optimal capacity scaling. We will show that, by proper network partitioning and link scheduling, network capacity scales on the same order as the theoretical results [1,7,12,34–41], but with significantly reduced scheduling complexity that can be as small as a constant. Before formulating the problem, we define a few relevant concepts.

***Definition* 3.1.** *Asymptotic bounds. (i) $f(n)=O(g(n))$ means there exists a constant c such that $f(n)\leq cg(n)$ as $n\to\infty$; (ii) $f(n)=o(g(n))$ means $\lim_{n\to\infty}\frac{f(n)}{g(n)}=0$; (iii) $f(n)=\Omega(g(n))$ means $g(n)=O(f(n))$; (iv) $f(n)=\omega(g(n))$ means $g(n)=o(f(n))$; (v) $f(n)=\Theta(g(n))$ means $f(n)=O(g(n))$ and $f(n)=\Omega(g(n))$. (See [12].)*

***Definition* 3.2.** *Scheduling diameter. Given a scheduling algorithm $\mathcal{S}(t)=\{l_{ij}(t):l_{ij}$ is active at time $t\}$, the scheduling diameter is defined as $\Phi(\mathcal{S}(t))=\max_{\{l_{i_1j_1},l_{i_2j_2}\in\mathcal{S}(t)\}}\{|X_{i_1}-X_{i_2}|\}$.*

***Definition* 3.3.** *Scheduling localization. In extended networks, algorithm $\mathcal{S}(t)$ is global if $\Phi(\mathcal{S}(t))=\Theta(\sqrt{n})$ and algorithm $\mathcal{S}(t)$ is localized if $\Phi(\mathcal{S}(t))=o(\sqrt{n})$.*

***Definition* 3.4.** *Scheduling partition. A scheduling partition is a geographic region in which the complete topology information is collectable for collision-free link scheduling. We*

consider convex polygonal partitions only.

***Definition* 3.5.** *Scheduling complexity. We consider a centralized scheduling algorithm in each partition. The scheduling complexity is the number of computational steps required in one run of the algorithm.*

All the previous scheduling algorithms assumed in the capacity scaling study [1,7,12,34–41] are global, thus facing the implementability problem in large-scale networks. Our objective is to achieve the same order optimal capacity with manageable scheduling complexity. By taking the partition approach, the scheduling task is completed independently in individual partitions, thus avoiding network-wide topology collection or information exchange. The challenge for the partition approach is the simultaneous satisfaction of two goals: maximum capacity (at least in the order sense) and minimum complexity (constant, if possible). Our scheduling partition problem is hence formulated as follows.

***Definition* 3.6.** *Network capacity. A data rate $\lambda(n)$ is the network capacity if there exists a joint packet routing and link scheduling scheme such that every node in the network can send data at rate $\lambda(n)$ to destinations losslessly, but not with any rate higher than $\lambda(n)$.*

Note that $\lambda(n)$ is determined jointly by the packet routing protocol and the link scheduling algorithm. As our research focus is the possibility of using localized scheduling algorithms to achieve order optimal capacity, we assume that the same routing protocol (the specific protocol selection is insignificant here) is applied when we compare the performance of global and localized scheduling algorithms.

Denoting $\lambda_g(n)$ as the $\lambda(n)$ when *any* of the global scheduling algorithms assumed in [1,7,12,34–41] is used, and $\lambda_l(n)$ as the $\lambda(n)$ when we partition the network and implement a localized scheduling algorithm in each partition, we will prove

$$\lambda_l(n) = \Theta(\lambda_g(n)) \tag{3.6}$$

is possible if we partition the network and design the localized scheduling algorithms appropriately. In other words, given $\lambda_g(n) = \Theta(f(n))$, we will demonstrate the way to achieve $\lambda_l(n) = \Theta(f(n))$, regardless of the function $f(n)$ that depends on the particular assumptions on the interference, location and communication models.

3.3 Main Results on Network Capacity

The key question in this study is *whether* and *how* scheduling partition can achieve the order optimal capacity in large-scale wireless networks. To facilitate our study, we characterize the scheduling partition methodology by using three parameters and define a class of localized scheduling algorithms $\{\mathcal{S}_{l(\rho,\delta,\xi)}(t)\}$ that satisfy three specifications.

S1: Each partition is a convex polygon disjoint from others. It contains an inner disk of radius $\rho(n)$ and is contained in an outer co-centric disk of radius $\sigma\rho(n)$ (constant $\sigma > 1$). The disk center is defined as the center of the partition.

S2: In each partition, the links (excluding those in the same group of broad/multi-cast branches) scheduled for concurrent transmissions must have a minimum interspace $\delta(n)$, which is defined as the Euclidean distance between the transmitters.

S3: Given any location in a partition, at any time at least one link must be scheduled for transmission within radius $\xi(n)$ from that location. The location of a link is the location of the transmitter.

The above specifications prescribe the partition dimension via $\rho(n)$ and the density of concurrent transmissions via $\delta(n)$ and $\xi(n)$. In practice, a large network can be partitioned using Voronoi tessellation [56] by placing a set of schedulers at strategic locations. Each node in the network contacts its nearest scheduler and follows the scheduling instructions from the chosen scheduler. When $\xi(n) > \delta(n)$, **S2** and **S3** can co-exist without conflict. According to Definition 3.4, $\mathcal{S}_{l(\rho,\delta,\xi)}(t)$ guarantees collision-free link transmissions within each individual partition, but does not preclude transmission failures due to cross-partition interference. Our results will demonstrate that the impact of cross-partition interference can be bounded in a way that the network capacity scales on the same order as using global link scheduling. For conciseness, we abbreviate $\mathcal{S}_{l(\rho,\delta,\xi)}(t)$ as $\mathcal{S}_l(t)$ from now on, and denote any global scheduling algorithm as $\mathcal{S}_g(t)$.

Next, we provide the sufficient and necessary conditions on $\rho(n)$, $\delta(n)$ and $\xi(n)$ for $\mathcal{S}_l(t)$ to achieve order optimal capacity. The following theorem summarizes our main result.

***Theorem* 3.1.** *In large wireless networks,* $\forall\mathbb{I} \in \{\mathbb{I}_{\text{prot}}, \mathbb{I}_{\text{phy}}, \mathbb{I}_{\text{gen}}\}$, $\forall\mathbb{L} \in \{\mathbb{L}_{\text{rand}}, \mathbb{L}_{\text{arbi}}\}$, $\forall\mathbb{C} \in \{\mathbb{C}_{\text{uni}}, \mathbb{C}_{\text{bro}}, \mathbb{C}_{\text{mul}}\}$, *a localized scheduling algorithm* $\mathcal{S}_l(t)$ *achieves* $\lambda_l(n) = \Theta(\lambda_g(n))$ *if and only if the following conditions are satisfied:*

- *Partition dimension:* $\rho(n) = \Omega(r(n))$*, the partition size should scale at least on the same order of the critical transmission radius* $r(n)$*.*
- *Minimum link separation:* $\delta(n) = O(r(n))$*, the minimum separation between concurrent transmissions sh-ould scale at most on the same order of the critical transmission radius* $r(n)$*.*
- *Maximum link separation:* $\xi(n) = \Omega(r(n))$*, the maximum separation between concurrent transmissions sh-ould scale at least on the same order of the critical transmission radius* $r(n)$*.*

Theorem 3.1 reveals that, in order for $\mathcal{S}_l(t)$ to achieve the order optimal capacity, the smallest acceptable partition dimension is $\Theta(r(n))$, which equals $\Theta(\sqrt{\log n})$ for random node locations and $\Theta(1)$ for arbitrary node locations. As $\mathcal{S}_g(t)$ is a special case of $\mathcal{S}_l(t)$ with partition size $\Theta(\sqrt{n})$, Theorem 3.1 demonstrates a significant reduction of scheduling complexity by using partition, while still allows $\mathcal{S}_l(t)$ to achieve the same order optimal capacity as $\mathcal{S}_g(t)$. In addition, our result also indicates $r(n)$ as the correct order of link separation for optimal capacity. Scheduling lower (i.e., $\delta(n) = \omega(r(n))$) or higher (i.e., $\xi(n) = o(r(n))$) density of concurrent transmissions will yield less network capacity by either wasting opportunities for parallel communications or introducing excessive link collisions.

Theorem 3.1 in fact consists of two statements regarding the capacity comparison of localized and global scheduling algorithms under conditions $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$:

P1: $\lambda_l(n) = O(\lambda_g(n))$, the localized algorithms cannot ach-ieve higher capacity than the global algorithms;

P2: $\lambda_l(n) = \Omega(\lambda_g(n))$, the localized algorithms can however achieve at least a constant fraction of capacity of the global algorithms.

We further have the following theorems that prove the above statements in Theorem 3.1.

Theorem 3.2. *Given a localized scheduling algorithm* $\mathcal{S}_l(t)$ *that achieves capacity* $\lambda_l(n)$*, there exists a global scheduling algorithm that achieves* $\lambda_g(n) \geq \lambda_l(n)$*.*

Proof. Suppose the network $\mathcal{B}$ is divided into a set of disjoint partitions $\{\mathcal{C}_i\}$ ($\mathcal{B} = \cup_i \mathcal{C}_i$) and there is a local schedule $\mathcal{S}_{l,i}(t)$ in each partition $\mathcal{C}_i$. Let $\widetilde{\mathcal{S}}_{l,i}(t)$ denote the subset of failed links in $\mathcal{S}_{l,i}(t)$ due to cross-partition collisions. Note that $\widetilde{\mathcal{S}}_{l,i}(t) = \emptyset$ in the generalized physical interference model as there is no absolute link failure in this model. By defining a global scheduling scheme $\mathcal{S}_g(t) = \cup_i(\mathcal{S}_{l,i}(t) \setminus \widetilde{\mathcal{S}}_{l,i}(t))$, we see that $\mathcal{S}_g(t)$ supports the same $\lambda_l(n)$ as $\mathcal{S}_{l,i}(t)$ using the same routing protocol. □

The proof of Theorem 3.2 is also the proof for statement **P1** of Theorem 3.1. To prove statement **P2**, we have the next theorem that transforms the problem on network capacity measured by the maximum source data rate to an equivalent problem on the data transmission rate on each wireless link in the network. The latter problem is easier to solve.

***Theorem* 3.3.** *Denoting $w_{l,ij}(n)$ and $w_{g,ij}(n)$ as the data transmission rates supported on link l_{ij} with localized and global scheduling respectively, $\lambda_l(n) = \Omega(\lambda_g(n))$ if and only if $w_{l,ij}(n) = \Omega(w_{g,ij}(n))$ for any l_{ij}.*

Proof. We define an indicator function $I_{l_{ij}}(\lambda_s)$ such that $I_{l_{ij}}(\lambda_s) = 1$ if the rate λ_s from source v_s traverses l_{ij} and $I_{l_{ij}}(\lambda_s) = 0$ otherwise. Obviously $w_{ij}(n) = \sum_{s=1}^{n} \lambda_s I_{l_{ij}}(\lambda_s)$, which shows that the data transmission rate on a link is the sum of all the traversing source rates. If we scale every source rate simultaneously by a constant, the data rate on every link in the network is scaled by the same constant.

We first prove the sufficient condition. If $w_{l,ij}(n) = \Omega(w_{g,ij}(n))$, there then exists a constant c that satisfies $w_{l,ij}(n) \geq c w_{g,ij}(n)$ as $n \to \infty$. It indicates that if we scale each source rate from $\lambda_g(n)$ to $c\lambda_g(n)$, $w_{l,ij}(n)$ can accommodate the new source rate. Thus, $\lambda_l(n) = \Omega(\lambda_g(n))$.

We next prove the necessary condition. When $\lambda_l(n) = \Omega(\lambda_g(n))$, then there exists a constant c such that, given $\lambda_g(n)$, localized scheduling can support $c\lambda_g(n)$. As the routing protocol is fixed, it is then required on every link l_{ij} that localized scheduling can accommodate the link transmission rate $c w_{g,ij}(n)$. Hence $w_{l,ij}(n) = \Omega(w_{g,ij}(n))$. □

Given Theorem 3.3, the problem described in statement **P2** becomes a comparison of $w_{l,ij}(n)$ and $w_{g,ij}(n)$, on which we have the following result.

***Theorem* 3.4.** *In large-scale wireless networks, $\forall \mathbb{I}, \forall \mathbb{L}, \forall \mathbb{C}$, a localized link scheduling algorithm $\mathcal{S}_l(t)$ achieves $w_{l,ij}(n) = \Omega(w_{g,ij}(n))$ for any link l_{ij} if and only if $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$.*

Theorems 3.2, 3.3 and 3.4 together constitute the proof for Theorem 3.1. In the next two sections, we will present the proof for Theorem 3.4, which includes the *sufficiency* and the *necessity* of conditions $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$ for obtaining $w_{l,ij}(n) = \Omega(w_{g,ij}(n))$. The main idea of proof is to compare the respective bounds for $w_{l,ij}(n)$ and $w_{g,ij}(n)$ under these scheduling conditions.

3.4 Sufficient Condition for Maximum Capacity

We prove in this section the sufficient condition in Theorem 3.4, which states that $w_{l,ij}(n) = \Omega(w_{g,ij}(n))$ if $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$. Our proof has two steps: i) we bound the link transmission rate of $\mathcal{S}_g(t)$ as $w_{g,ij}(n) = \Theta(\frac{1}{r^2(n)})$, and ii) we bound the corresponding link transmission rate of $\mathcal{S}_l(t)$ as $w_{l,ij}(n) = \Omega(\frac{1}{r^2(n)})$. The combination immediately gives $w_{l,ij}(n) = \Omega(w_{g,ij}(n))$.

3.4.1 Link Rate with Global Scheduling

We have the following theorem on bounding the link transmission rate in global scheduling.

***Theorem* 3.5.** *In large-scale wireless networks, $\forall\mathbb{I}, \forall\mathbb{L}, \forall\mathbb{C}$, the maximum achievable link transmission rate in global scheduling is $w_{g,ij}(n) = \Theta(\frac{1}{r^2(n)})$.*

Proof. We discuss the interference models $\mathbb{I}_{\text{prot}}$, $\mathbb{I}_{\text{phy}}$ and $\mathbb{I}_{\text{gen}}$ separately. As this proof does not make any assumption regarding node locations and communication types, it applies to all the models of $\mathbb{L}$ and $\mathbb{C}$. The foundation of this proof is the fact that links do not collide by using $\mathcal{S}_g(t)$.

For $\mathbb{I}_{\text{prot}}$, any two concurrently transmitting nodes must be separated at least by distance $\Delta r(n)$. Otherwise, both transmissions fail due to the violation of Eq. (3.2). Therefore, at any time $\mathcal{S}_g(t)$ can schedule at most one sender in any arbitrary square region with edge $\frac{\Delta r(n)}{\sqrt{2}}$. Given node density 1, we bound the time-average data transmission rate of link l_{ij} as $w_{g,ij}(n) \leq \eta_{ij}\frac{W}{(\Delta r(n)/\sqrt{2})^2} = \frac{2\eta_{ij}W}{\Delta^2 r^2(n)}$. We obtain the lower bound on $w_{g,ij}(n)$ by observing that senders spaced by $(2+\Delta)r(n)$ do not collide, i.e., $w_{g,ij}(n) \geq \frac{\eta_{ij}W}{(2+\Delta)^2 r^2(n)}$.

For $\mathbb{I}_{\text{phy}}$, a transmission is successful if Eq. (3.3) holds. As shown in Fig. 3.1, when v_i sends a packet to v_j, the number of concurrent senders y within distance $\frac{1}{2}|X_i - X_j|$ from

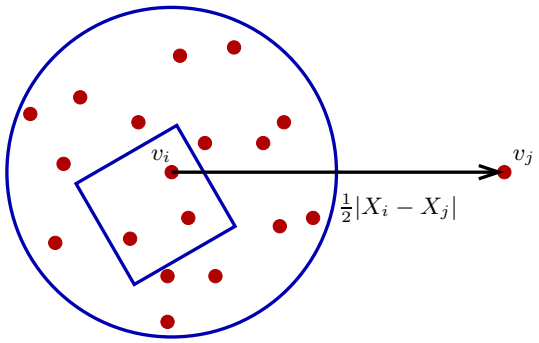


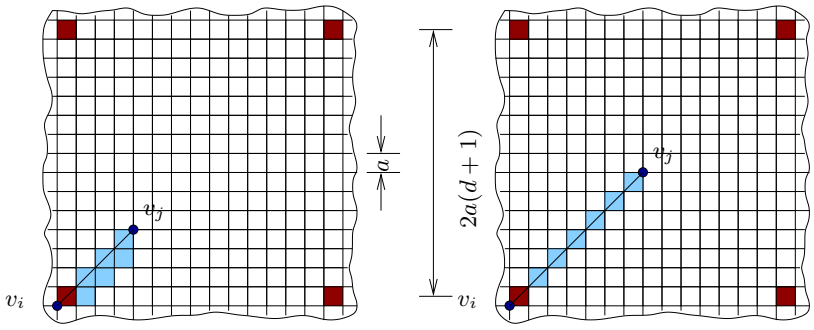
Figure 3.1: Density determination of the concurrent senders in the physical model. The disk has radius $\frac{1}{2}|X_i - X_j|$. The square has edge length $\frac{1}{2\sqrt{2}}|X_i - X_j|$.

v_i must satisfy the following inequality in order for v_j to receive the packet correctly

$$\frac{P_{\max}}{|X_i - X_j|^\alpha} \geq \beta y \frac{P_{\min}}{(\frac{3}{2}|X_i - X_j|)^\alpha}, \tag{3.7}$$

from which we obtain $y \leq \frac{(3/2)^\alpha P_{\max}}{\beta P_{\min}}$. By taking an arbitrary square region with edge $\frac{1}{2\sqrt{2}}|X_i - X_j|$ that covers v_i, as illustrated in Fig. 3.1, and noting that $|X_i - X_j| \geq r_0(n)$, we show that the number of concurrent senders within any square region with edge $\frac{r_0(n)}{2\sqrt{2}}$ cannot exceed $\frac{(3/2)^\alpha P_{\max}}{\beta P_{\min}} + 1$. Otherwise, all these transmissions fail due to their mutual interference. Therefore, we have $w_{g,ij}(n) \leq \eta_{ij} \frac{(3/2)^\alpha P_{\max}/(\beta P_{\min})+1}{(r_0(n)/(2\sqrt{2}))^2} W = \frac{8((3/2)^\alpha P_{\max}/(\beta P_{\min})+1)\eta_{ij} W}{\varepsilon^2 r^2(n)}$. To find the lower bound on $w_{g,ij}(n)$, we cite a relevant result. It is pointed out in [1] that there is a large enough constant Δ_1 such that all the transmissions are successful in the physical model as long as the concurrent senders are separated by distance $(2 + \Delta_1)r(n)$. We thus obtain easily $w_{g,ij}(n) \geq \frac{\eta_{ij} W}{(2+\Delta_1)^2 r^2(n)}$.

For $\mathbb{I}_{\text{gen}}$, by dividing the network into small cells with edge a, link l_{ij} can transmit successfully at rate $R(a, d)$ when scheduled [7, 40], where nodes v_i and v_j are separated by Manhattan distance d and function $R(a, d)$ is a constant under our assumptions on $\mathbb{I}_{\text{gen}}$. The Manhattan distance is defined as the minimum number of contiguous cells to connect v_i and v_j. Two cells are contiguous in [7] if they share an edge, while in [40] they are contiguous if they share either an edge or a vertex. We consider both definitions here. We find that the critical node transmission radius $r(n)$ must satisfy the following relations with the cell size a and the Manhattan distance d: $(\frac{d-1}{2} + 1)\sqrt{2}a \leq r(n) \leq (d + 1)\sqrt{2}a$. As illustrated in the fraction of network in Fig. 3.2, we assume that a node transmits in each of the four corner cells that are separated by Euclidean distance $2a(d + 1)$ as required in [7] and [40]. Considering all the possibilities of d (regardless of its parity) and the



(a) $r(n) \geq (\frac{d-1}{2}+1)\sqrt{2}a$ (b) $r(n) \leq (d+1)\sqrt{2}a$

Figure 3.2: Determination of the bounds on $r(n)$ in the generalized physical model.

arbitrary locations of v_i and v_j in their respective cells, $r(n) \geq (\frac{d-1}{2}+1)\sqrt{2}a$ is necessary to guarantee the existence of a connection between v_i and v_j. Besides, given Manhattan distance d, $r(n) \leq (d+1)\sqrt{2}a$ since the Euclidean distance between v_i and v_j cannot exceed $(d+1)\sqrt{2}a$. Because at any time the scheduling algorithm $\mathcal{S}_g(t)$ presented in [7] and [40] allows only one node to transmit within any square region of area $4a^2(d+1)^2$ and the bounds on $r(n)$ give $2r^2(n) \leq 4a^2(d+1)^2 \leq 8r^2(n)$, we have $\frac{\eta_{ij}R(a,d)}{8r^2(n)} \leq w_{g,ij}(n) \leq \frac{\eta_{ij}R(a,d)}{2r^2(n)}$.

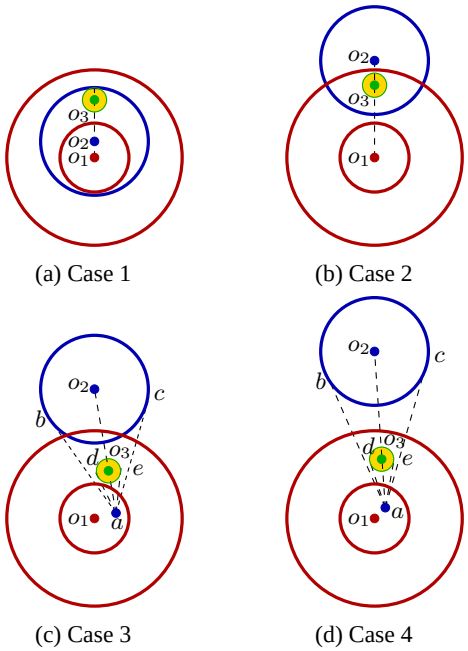
We define a constant $c_1 = \min\{\frac{\eta_{ij}W}{(2+\Delta)^2}, \frac{\eta_{ij}W}{(2+\Delta_1)^2}, \frac{\eta_{ij}R(a,d)}{8}\}$ and a constant $c_2 = \max\{\frac{2\eta_{ij}W}{\Delta^2}, \frac{8((3/2)^\alpha P_{\max}/(\beta P_{\min})+1)\eta_{ij}W}{\varepsilon^2}, \frac{\eta_{ij}R(a,d)}{2}\}$, we summarize the bounds as $\frac{c_1}{r^2(n)} \leq w_{g,ij}(n) \leq \frac{c_2}{r^2(n)}$ in all the three interference models. Thus, $w_{g,ij}(n) = \Theta(\frac{1}{r^2(n)})$. □

3.4.2 Link Rate with Localized Scheduling

We next determine the lower bound on the link transmission rate when $\mathcal{S}_l(t)$ is used. Before presenting the theorem on this lower bound, we have the following two lemmas that will be used in the proof of the theorem.

***Lemma* 3.1.** *Assume that $\kappa_1, \kappa_2 > 0$ are constants. Given a disk of radius $\kappa_1 r(n)$, if $\rho(n) \geq \kappa_2 r(n)$, the number of partitions that overlap the disk is at most a constant.*

Proof. Since specification **S1** dictates that the partition diameter is not larger than $2\sigma\rho(n)$, any partition that overlaps with the disk must be contained in a larger co-centric disk of radius $\kappa_1 r(n) + 2\sigma\rho(n)$. Racall that the partitions are disjoint and each occupies at least an area of $\pi\rho^2(n)$. Hence, the number of partitions that can be contained in the larger disk is at most $\frac{\pi(\kappa_1 r(n)+2\sigma\rho(n))^2}{\pi\rho^2(n)} \leq (\frac{\kappa_1}{\kappa_2}+2\sigma)^2$, which is a constant. □



(a) Case 1 (b) Case 2

(c) Case 3 (d) Case 4

Figure 3.3: The relative locations of the co-centric disks and an overlapping partition. The co-centric disks are centered at o_1, denoted as $D(o_1, \kappa_1 r)$ and $D(o_1, \kappa_3 r)$ respectively. The annulus is denoted as $D(o_1, \kappa_3 r) \setminus D(o_1, \kappa_1 r)$. The partition is centered at o_2. For clarity, we have only shown the inner disk of radius ρ, denoted as $D(o_2, \rho)$, but ignored the partition boundary as well as the outer disk of radius $\sigma\rho$. Point a is inside $D(o_1, \kappa_1 r)$. Lines $\overline{ab}$ and $\overline{ac}$ are tangent to $D(o_2, \rho)$. Lines $\overline{ad}$ and $\overline{ae}$ are tangent to $D(o_3, \epsilon)$. $d = |\overline{o_1 o_2}|$.

***Lemma* 3.2.** *Assume that constants κ_1, κ_2 and κ_3 satisfy $0 < \kappa_1 < \kappa_2 < \kappa_3 \leq 2\kappa_1$. Given two co-centric disks of radius $\kappa_1 r(n)$ and $\kappa_3 r(n)$ respectively, define an annulus as the area between the disk circumferences. If $\rho(n) \geq \kappa_2 r(n)$, there exists a constant $c > 0$ such that a partition that overlaps with the smaller disk must overlap with the annulus by at least an area of $cr^2(n)$.*

Proof. For simplicity reason, we abbreviate $r(n)$ and $\rho(n)$ as r and ρ respectively in this proof. Besides, we use $\mathcal{O}$ to denote the overlap area between a partition and the annulus. The idea of the proof is to find a disk in $\mathcal{O}$, denoted as $D(o_3, \epsilon)$ where o_3 is the center and ϵ is the radius, which covers an area of at least cr^2. By defining d as the Euclidean distance between the center of the two co-centric disks and the center of an overlapping partition, we discuss all the possible cases of d.

Case 1: $0 \leq d \leq \kappa_3 r - \kappa_2 r$. As shown in Fig. 3.3(a), we find point o_3 on the extension of line $\overline{o_1 o_2}$ such that $|\overline{o_2 o_3}| = \frac{\kappa_1 r + \kappa_2 r}{2}$ and define $\epsilon = \frac{\kappa_2 r - \kappa_1 r}{2}$. Given any point $p \in D(o_3, \epsilon)$, it is easy to see that $|\overline{o_2 p}| \leq |\overline{o_2 o_3}| + |\overline{o_3 p}| \leq \frac{\kappa_1 r + \kappa_2 r}{2} + \frac{\kappa_2 r - \kappa_1 r}{2} = \kappa_2 r \leq \rho$,

indicating $p \in D(o_2, \rho)$. In addition, $|\overline{o_1p}| \geq |\overline{o_1o_3}| - |\overline{o_3p}| \geq \frac{\kappa_1 r+\kappa_2 r}{2} - \frac{\kappa_2 r-\kappa_1 r}{2} = \kappa_1 r$ and $|\overline{o_1p}| \leq |\overline{o_1o_3}| + |\overline{o_3p}| \leq \kappa_3 r - \kappa_2 r + \frac{\kappa_1 r+\kappa_2 r}{2} + \frac{\kappa_2 r-\kappa_1 r}{2} = \kappa_3 r$, showing $p \in D(o_1, \kappa_3 r) \setminus D(o_1, \kappa_1 r)$. Hence, $D(o_3, \epsilon) \subseteq \mathcal{O}$. Since the area of $D(o_3, \epsilon)$ is at least $\pi(\frac{\kappa_2 r-\kappa_1 r}{2})^2$, we have proven the lemma by choosing $c_1 = \frac{\pi}{4}(\kappa_2 - \kappa_1)^2$.

Case 2: $\kappa_3 r - \kappa_2 r < d \leq \rho + \frac{\kappa_1 r+\kappa_3 r}{2}$. As shown in Fig. 3.3(b), $D(o_2, \rho)$ overlaps with $D(o_1, \kappa_3 r)$ partially. We locate point o_3 on line $\overline{o_1o_2}$ such that $|\overline{o_1o_3}| = \frac{\kappa_1 r+3\kappa_3 r}{4}$ and define $\epsilon = \frac{\kappa_3 r-\kappa_1 r}{4}$. It is obvious that $D(o_3, \epsilon) \subseteq D(o_1, \kappa_3 r) \setminus D(o_1, \kappa_1 r)$. Besides, we obtain $|d - \frac{\kappa_1 r+3\kappa_3 r}{4}| \leq \rho - \frac{\kappa_3 r-\kappa_1 r}{4}$ from $\kappa_3 r - \kappa_2 r < d \leq \rho + \frac{\kappa_1 r+\kappa_3 r}{2}$. Given any point $p \in D(o_3, \epsilon)$, we have $|\overline{o_2p}| \leq |\overline{o_2o_3}| + |\overline{o_3p}| \leq |d - \frac{\kappa_1 r+3\kappa_3 r}{4}| + \frac{\kappa_3 r-\kappa_1 r}{4} \leq \rho - \frac{\kappa_3 r-\kappa_1 r}{4} + \frac{\kappa_3 r-\kappa_1 r}{4} = \rho$, indicating $D(o_3, \epsilon) \subseteq D(o_2, \rho)$. Thus, $D(o_3, \epsilon) \subseteq \mathcal{O}$ and $D(o_3, \epsilon)$ has an area $\pi(\frac{\kappa_3 r-\kappa_1 r}{4})^2$. The lemma is true by choosing $c_2 = \frac{\pi}{16}(\kappa_3 - \kappa_1)^2$.

Case 3: $\rho + \frac{\kappa_1 r+\kappa_3 r}{2} < d \leq \rho + \kappa_3 r$. As shown in Fig. 3.3(c), $D(o_2, \rho)$ overlaps with $D(o_1, \kappa_3 r)$ partially in this case too. Let a denote a point over which the partition overlaps with $D(o_1, \kappa_1 r)$, $\overline{ab}$ and $\overline{ac}$ denote the lines tangent to $D(o_2, \rho)$. Because the partition is convex, the area encompassed by $\overline{ab}$, $\overline{ac}$ and $D(o_2, \rho)$ must be part of the partition too. We choose point o_3 on line $\overline{o_2a}$ such that $|\overline{o_1o_3}| = \frac{3\kappa_1 r+\kappa_3 r}{4}$, which is possible because $|\overline{o_1a}| < \kappa_1 r$, $|\overline{o_1o_2}| > \frac{\kappa_1 r+\kappa_3 r}{2}$ and $\overline{o_2a}$ is continuous. We then define $\epsilon = \frac{\rho(\kappa_3 r-\kappa_1 r)}{8(\rho+\kappa_1 r+\kappa_3 r)}$. It is obvious that $D(o_3, \epsilon) \subseteq D(o_1, \kappa_3 r) \setminus D(o_1, \kappa_1 r)$. To prove $D(o_3, \epsilon)$ is inside the partition, it is only necessary to show $\angle o_3ad < \angle o_2ab$, where $\overline{ad}$ is tangent to $D(o_3, \epsilon)$. Since $\sin(\angle o_3ad) = |\overline{o_3d}|/|\overline{o_3a}| \leq \frac{\epsilon}{(3\kappa_1 r+\kappa_3 r)/4-\kappa_1 r} < \frac{\rho}{\rho+\kappa_1 r+\kappa_3 r} \leq |\overline{o_2b}|/|\overline{o_2a}| = \sin(\angle o_2ab)$, we obtain $\angle o_3ad < \angle o_2ab$. Hence, $D(o_3, \epsilon) \subseteq \mathcal{O}$. The area of $D(o_3, \epsilon)$ is at least $\pi(\frac{\kappa_2 r(\kappa_3 r-\kappa_1 r)}{8(\kappa_1 r+\kappa_2 r+\kappa_3 r)})^2$, so the lemma is true by choosing $c_3 = \frac{\pi}{64}(\frac{\kappa_2(\kappa_3-\kappa_1)}{\kappa_1+\kappa_2+\kappa_3})^2$.

Case 4: $\rho + \kappa_3 r < d \leq \sigma\rho + \kappa_1 r$. Note that $\rho + \kappa_3 r < \sigma\rho + \kappa_1 r$ when the partition overlaps with $D(o_1, \kappa_1 r)$ and $d > \rho + \kappa_3 r$. The proof is similar to Case 3. As shown in Fig. 3.3(d), we locate point o_3 on line $\overline{o_2a}$ such that $|\overline{o_1o_3}| = \frac{\kappa_1 r+\kappa_3 r}{2}$ and define $\epsilon = \frac{\rho(\kappa_3 r-\kappa_1 r)}{4(\sigma\rho+2\kappa_1 r)}$. It is obvious that $D(o_3, \epsilon) \subseteq D(o_1, \kappa_3 r) \setminus D(o_1, \kappa_1 r)$. Again, we define $\overline{ab}$ and $\overline{ac}$ to be tangent to $D(o_2, \rho)$, $\overline{ad}$ and $\overline{ae}$ to be tangent to $D(o_3, \epsilon)$. We have $\angle o_3ad < \angle o_2ab$ because $\sin(\angle o_3ad) = |\overline{o_3d}|/|\overline{o_3a}| \leq \frac{\epsilon}{(\kappa_1 r+\kappa_3 r)/2-\kappa_1 r} < \frac{\rho}{\sigma\rho+2\kappa_1 r} \leq |\overline{o_2b}|/|\overline{o_2a}| = \sin(\angle o_2ab)$. So, $D(o_3, \epsilon) \subseteq \mathcal{O}$. Proof follows using $c_4 = \frac{\pi}{16}(\frac{\kappa_2(\kappa_3-\kappa_1)}{2\kappa_1+\sigma\kappa_2})^2$.

For any $d > \sigma\rho + \kappa_1 r$, the partition does not overlap with $D(o_1, \kappa_1 r)$. We have therefore considered all the possible cases regarding d. In summary, the lemma is proven true by choosing $c = \min\{c_1, c_2, c_3, c_4\}$. □

It is worth noted that Lemma 3.2 is still true if $\kappa_3 > 2\kappa_1$. We do not elaborate since the current form of Lemma 3.2 is enough for us to prove a lower bound on $w_{l,ij}(n)$.

***Theorem* 3.6.** *In large-scale wireless networks, $\forall\mathbb{I}, \forall\mathbb{L}, \forall\mathbb{C}$, a localized link scheduling algorithm $\mathcal{S}_l(t)$ achieves $w_{l,ij}(n) = \Omega(\frac{1}{r^2(n)})$ for any link l_{ij} if $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$.*

Proof. By the definition of asymptotic bounds, we need to find constants c, c_ρ, c_δ and c_ξ such that $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ if $\rho(n) \geq c_\rho r(n)$, $\delta(n) \leq c_\delta r(n)$ and $\xi(n) \geq c_\xi r(n)$. We present the proof for $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ in two steps:

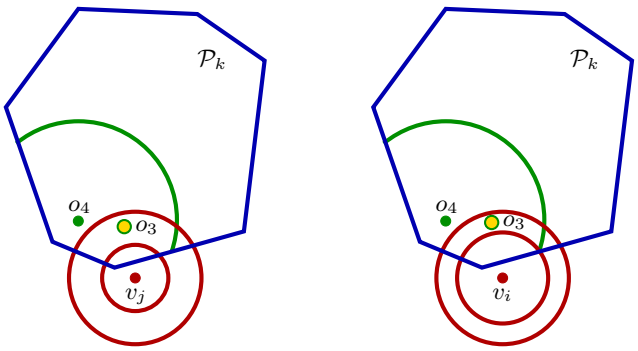
1: $\rho(n) = c_\rho r(n)$, $\delta(n) = c_\delta r(n)$, $\xi(n) = c_\xi r(n)$.

2: $\rho(n) \geq c_\rho r(n)$, $\delta(n) \leq c_\delta r(n)$, $\xi(n) \geq c_\xi r(n)$.

Step 1 is the special case in which we show that $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ when we configure $\rho(n)$, $\delta(n)$ and $\xi(n)$ correctly, and step 2 is the generalized case in which $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ is still true if $\rho(n)$ and $\xi(n)$ are configured larger while $\delta(n)$ is configured smaller than our choices. Next, we present the proof for all the models of $\mathbb{I}$, $\mathbb{L}$ and $\mathbb{C}$.

Step 1, for $\mathbb{I}_{\text{prot}}$. We first consider location model $\mathbb{L}_{\text{rand}}$ and communication model $\mathbb{C}_{\text{uni}}$. We demonstrate that if we choose constants $c_\rho = \frac{3}{2}(1+\Delta)$, $c_\delta = 3(1+\Delta)$, $c_\xi = 4(1+\Delta)$ and some constant c_u, we will have $w_{l,ij}(n) \geq \frac{c_u}{r^2(n)}$ when $\rho(n) = c_\rho r(n)$, $\delta(n) = c_\delta r(n)$ and $\xi(n) = c_\xi r(n)$. By $\mathbb{I}_{\text{prot}}$, a transmission from v_i to v_j is successful when there are no other concurrent senders within distance $(1+\Delta)r(n)$ from v_j. Since $\mathcal{S}_l(t)$ guarantees non-collision within individual partitions, any sender that fails the transmission on link l_{ij}, if any, must reside in a neighbor partition that overlaps with disk $D(v_j, (1+\Delta)r(n))$. By setting $\kappa_1 = 1+\Delta$ and $\kappa_2 = \frac{3}{2}(1+\Delta)$, Lemma 3.1 shows that $D(v_j, (1+\Delta)r(n))$ overlaps with at most a constant c_1 number of partitions. In addition, by setting $\kappa_3 = 2(1+\Delta)$, Lemma 3.2 tells us that each of these overlapping partitions must overlap with the annulus $D(v_j, 2(1+\Delta)r(n)) \setminus D(v_j, (1+\Delta)r(n))$ at least by an area $c_2 r^2(n)$, where c_2 is a constant, as depicted in Fig. 3.4(a). In any neighbor partition, for example $\mathcal{P}_k$, because specification **S3** dictates at least one active sender within any arbitrarily located disk of radius $\xi(n)$, for example $D(o_4, 4(1+\Delta)r(n))$, the probability of finding an active sender inside $D(o_3, \epsilon)$ is bounded as

$$\Pr[\text{active sender in } D(o_3, \epsilon)] \geq \frac{c_2 r^2(n)}{\pi(4(1+\Delta)r(n))^2} = c_3. \tag{3.8}$$



(a) Unicast (b) Broadcast/Multicast

Figure 3.4: Proof of Theorem 3.6 in the protocol interference model. In unicast, we define disks $D(v_j, (1+\Delta)r(n))$, $D(v_j, 2(1+\Delta)r(n))$, $D(o_3, \epsilon)$ and $D(o_4, 4(1+\Delta)r(n))$. In broad/multi-cast, we define disks $D(v_i, (2+\Delta)r(n))$, $D(v_i, 2(1+\Delta)r(n))$, $D(o_3, \epsilon)$ and $D(o_4, 4(1+\Delta)r(n))$. The area of $D(o_3, \epsilon)$ is at least $c_2 r^2(n)$. The polygon denotes a neighbor partition.

Since $\delta(n) = 3(1+\Delta)r(n)$, when a node in $D(o_3, \epsilon)$ is active in transmission, $\mathcal{P}_k$ does not schedule any concurrent senders in $D(v_j, (1+\Delta)r(n))$. Hence we obtain

$$\Pr[\mathcal{P}_k \text{ does not fail reception at } v_j \text{ at a given time}] \geq c_3. \tag{3.9}$$

As $\mathcal{P}_k$ may reschedule up to $\lfloor \frac{L_{\max}}{L_{\min}} \rfloor + 2$ times during the packet transmission from v_i to v_j, we further have

$$\Pr[\mathcal{P}_k \text{ does not fail reception at } v_j] \geq c_3^{\lfloor \frac{L_{\max}}{L_{\min}} \rfloor + 2} = c_4. \tag{3.10}$$

Finally, considering all the neighbor partitions, we have

$$\Pr[\text{successful reception at } v_j] \geq c_4^{c_1} = c_5. \tag{3.11}$$

Note that at least one sender is active within any square region with edge $2\xi(n)$ and the node density is 1, so v_i has a chance of at least $\frac{1}{64(1+\Delta)^2 r^2(n)}$ to be scheduled. Thus, we can bound the time-average transmission rate of l_{ij} as

$$w_{l,ij}(n) \geq \frac{c_5 \eta_{ij} W}{64(1+\Delta)^2 r^2(n)} = \frac{c_u}{r^2(n)}, \tag{3.12}$$

where $c_u = \frac{c_5 \eta_{ij} W}{64(1+\Delta)^2}$.

Step 1, for $\mathbb{I}_{\text{prot}}$ (cont'd). We then consider location model $\mathbb{L}_{\text{rand}}$ with communication models $\mathbb{C}_{\text{bro}}$ and $\mathbb{C}_{\text{mul}}$. By $\mathbb{C}_{\text{bro}}$ and $\mathbb{C}_{\text{mul}}$, the packet transmissions from v_i

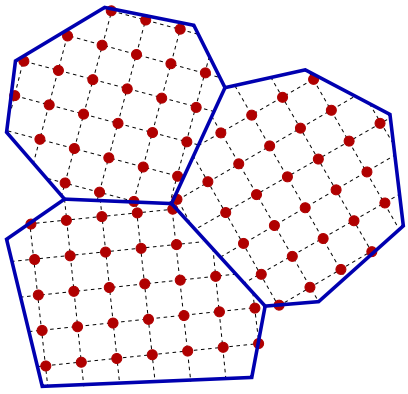


Figure 3.5: Node locations on grids.

along the broad/multi-cast branches are successful if there are no concurrent senders within distance $(2+\Delta)r(n)$ from v_i. Similar to unicast, we prove the existence of a constant c_6 such that

$$\Pr[\text{successful reception at all receivers}] \geq c_6, \tag{3.13}$$

by setting $\rho(n) = (2+\frac{3}{2}\Delta)r(n)$, $\delta(n) = (4+3\Delta)r(n)$, $\xi(n) = 4(1+\Delta)r(n)$, $\kappa_1 = 2+\Delta$, $\kappa_2 = 2+\frac{3}{2}\Delta$ and $\kappa_3 = 2(1+\Delta)$, as shown in Fig. 3.4(b). We further prove

$$w_{l,ij}(n) \geq \frac{c_6\eta_{ij}W}{64(1+\Delta)^2 r^2(n)} = \frac{c_b}{r^2(n)}, \tag{3.14}$$

where $c_b = \frac{c_6\eta_{ij}W}{64(1+\Delta)^2}$.

Combining the analysis of $\mathbb{C}_{\text{uni}}$, $\mathbb{C}_{\text{bro}}$, $\mathbb{C}_{\text{mul}}$ and defining $c = \min\{c_u, c_b\}$, we have proven $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ for random node locations in the protocol interference model.

Step 1, for $\mathbb{I}_{\text{prot}}$ (cont'd). We next consider location model $\mathbb{L}_{\text{arbi}}$ in which node locations are assigned in need. A popular assignment is to place the nodes on a grid with equal distance between neighbors [1, 38]. When we partition the network, we align the node locations into a grid form in each partition. The grids are however not aligned across partitions due to their independence from one another, as shown in Fig. 3.5. The node interspace in each partition is set to 1 to satisfy the constant node density of 1. Given an arbitrary point on the partition boundary, because there is at least one node within distance $\sqrt{2}$ in each neighbor partition, any constant $r(n) \geq 2\sqrt{2}$ is large enough to guarantee the network connectivity. The proof for $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ is almost the same as that for the $\mathbb{L}_{\text{rand}}$ model. For conciseness, we only highlight the differences. In Eq. (3.8), we have shown that the probability for a neighbor partition to schedule a sender in the annulus area is at least a

constant c_3, where c_3 is defined as the ratio between two areas: $D(o_3, \epsilon)$ versus $D(o_4, 4(1+\Delta)r(n))$. To be accurate, c_3 should be the ratio between the numbers of nodes therein instead of the areas. These two metrics are equivalent for random node locations since $r(n) = \Theta(\sqrt{\log n}) \to \infty$ as $n \to \infty$ and the constant node density can be equivalently scaled to infinity if $r(n)$ is normalized to 1. For arbitrary node locations, $r(n) = \Theta(1)$ and the two ratios are not interchangeable. We next determine the ratio between the numbers of nodes. Recall that the area of $D(o_3, \epsilon)$ is at least $c_2 r^2(n)$. By choosing $r(n) \geq \max\{\sqrt{\frac{\pi}{c_2}}, 2\sqrt{2}\}$, we show that $D(o_3, \epsilon)$ contains at least one node. In general, there are at least $\lfloor\sqrt{\frac{c_2}{\pi}} r(n)\rfloor^2$ nodes inside $D(o_3, \epsilon)$. As every node placed inside $D(o_4, 4(1+\Delta)r(n))$ occupies a unit square area in a slightly larger disk $D(o_4, 4(1+\Delta)r(n) + \frac{\sqrt{2}}{2})$, a partition can place at most $\pi(4(1+\Delta)r(n) + \frac{\sqrt{2}}{2})^2$ nodes in $D(o_4, 4(1+\Delta)r(n))$. By setting $c_3' = \frac{\lfloor\sqrt{\frac{c_2}{\pi}} r(n)\rfloor^2}{\pi(4(1+\Delta)r(n)+\frac{\sqrt{2}}{2})^2}$, where $r(n)$ is constant for arbitrary node locations, we can rewrite Eq. (3.8) as

$$\Pr[\text{active sender in } D(o_3, \epsilon)] \geq \frac{\lfloor\sqrt{\frac{c_2}{\pi}} r(n)\rfloor^2}{\pi(4(1+\Delta)r(n) + \frac{\sqrt{2}}{2})^2} = c_3'. \tag{3.15}$$

Following the same reasoning as (3.9), (3.10) and (3.11), we have

$$\Pr[\text{successful reception at } v_j] \geq c_5'. \tag{3.16}$$

Because at least one node is active in any square region with edge $2\xi(n)$, which contains no more than $(8(1+\Delta)r(n) + \sqrt{2})^2$ nodes, v_i has a chance of at least $\frac{1}{(8(1+\Delta)r(n)+\sqrt{2})^2}$ to be scheduled. We hence obtain

$$w_{l,ij}(n) \geq \frac{c_5' \eta_{ij} W}{(8(1+\Delta)r(n) + \sqrt{2})^2} = \frac{c_u'}{r^2(n)}, \tag{3.17}$$

where $c_u' = \frac{c_5' \eta_{ij} W r^2(n)}{(8(1+\Delta)r(n)+\sqrt{2})^2}$. Similarly, by setting $c_b' = \frac{c_6' \eta_{ij} W r^2(n)}{(8(1+\Delta)r(n)+\sqrt{2})^2}$ in broadcast and multicast, we have

$$w_{l,ij}(n) \geq \frac{c_6' \eta_{ij} W}{(8(1+\Delta)r(n) + \sqrt{2})^2} = \frac{c_b'}{r^2(n)}. \tag{3.18}$$

We have thus proven the case for arbitrary node locations in the protocol interference model by setting $c = \min\{c_u', c_b'\}$.

Step 1, for $\mathbb{I}_{\text{phy}}$. Similar to the proof technique used in [1], we transform the physical model into an equivalent protocol model. It is shown in [1] that every node can receive packets correctly in the physical model as long as a minimum space $(2+\Delta_1)r(n)$ is enforced between neighbor senders, where Δ_1 is a constant. Distance $(2+\Delta_1)r(n)$

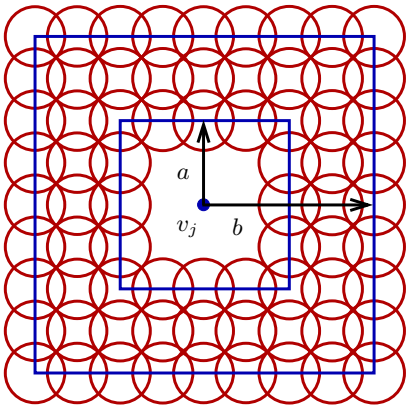


Figure 3.6: Aggregate interference from the region bounded between distance a and b.

demarcates *close-in* and *far-away* regions, and the aggregate interference from the far-away region is insignificant to correct packet reception. Similar result is also found in [42]. In the following, we extend this finding to network partition and prove the existence of another constant Δ_2. Specifically, we show that if $\rho(n) \geq r(n)$ and the concurrent senders in each partition are separated at least by $(2+\Delta_2)r(n)$, a transmission from v_i is successful if no other nodes transmit within radius $(2+\Delta_2)r(n)$ from v_i.

In the physical interference model, the SINR at a receiver v_j must exceed threshold β in order to receive a packet successfully. To determine the interference imposed on v_j, we divide the network into non-overlapping square belt regions and consider the aggregate interference from one of them, as illustrated in Fig. 3.6. We cover this square belt region seamlessly with disks of radius $\frac{(2+\Delta_2)r(n)}{2}$, which are interspaced evenly at distance $\frac{\sqrt{2}(2+\Delta_2)r(n)}{2}$. The disks needed is at most $(\frac{2b}{\sqrt{2}(2+\Delta_2)r(n)/2}+1)^2-(\frac{2a}{\sqrt{2}(2+\Delta_2)r(n)/2}-1)^2$, if $\frac{\sqrt{2}(2+\Delta_2)r(n)}{2}$ divides $2a$ and $2b$. Note that a partition cannot schedule more than one sender in a disk at any time because of the required minimum separation $(2+\Delta_2)r(n)$ between concurrent senders. The number of concurrent senders inside a disk thus does not exceed the number of partitions that overlap with this disk, which is at most $\frac{\pi((2+\Delta_2)r(n)/2+2\sigma\rho(n))^2}{\pi\rho^2(n)} \leq (\frac{2+\Delta_2}{2}+2\sigma)^2 \leq (2+\Delta_2)^2$. Here we have assumed $\Delta_2 \geq 4\sigma-2$. Letting $a = m\frac{\sqrt{2}(2+\Delta_2)r(n)}{4}$ and $b = (m+1)\frac{\sqrt{2}(2+\Delta_2)r(n)}{4}$, where $m = 1, 2, \cdots$, and noting that the interference from any sender inside the square belt zone does not exceed $P_{\max}a^{-\alpha}$, we bound the total interference

I from all the concurrent senders outside the square $a = \frac{\sqrt{2}(2+\Delta_2)r(n)}{4}$ as

$$\begin{aligned} I &\leq \sum_{m=1}^{\infty} 3(2m+1)(2+\Delta_2)^2 P_{\max}\Big(m\frac{\sqrt{2}(2+\Delta_2)r(n)}{4}\Big)^{-\alpha} \\ &\leq 3(2+\Delta_2)^2 P_{\max}\Big(\frac{\sqrt{2}(2+\Delta_2)r(n)}{4}\Big)^{-\alpha}\frac{3\alpha^2-6\alpha+2}{\alpha^2-3\alpha+2} \\ &= 3(2+\Delta_2)^{2-\alpha} P_{\max}\Big(\frac{\sqrt{2}r(n)}{4}\Big)^{-\alpha}\frac{3\alpha^2-6\alpha+2}{\alpha^2-3\alpha+2}. \end{aligned} \tag{3.19}$$

The SINR at v_j is then bounded by

$$\text{SINR} \geq \frac{P_{\min} r^{-\alpha}(n)}{BN_0 + 3(2+\Delta_2)^{2-\alpha} P_{\max}\big(\frac{\sqrt{2}r(n)}{4}\big)^{-\alpha}\frac{3\alpha^2-6\alpha+2}{\alpha^2-3\alpha+2}}. \tag{3.20}$$

After some computations, we show SINR $\geq \beta$ if

$$\Delta_2 \geq \Big(\frac{3(2\sqrt{2})^{\alpha}(3\alpha^2-6\alpha+2)\beta\gamma_1 P_{\max}}{(\alpha^2-3\alpha+2)(\gamma_1-\beta)P_{\min}}\Big)^{\frac{1}{\alpha-2}} - 2. \tag{3.21}$$

Eq. (3.21) demonstrates an important fact regarding localized scheduling in the physical interference model: if the concurrent senders in each partition are separated at least by $(2+\Delta_2)r(n)$ and constant $\Delta_2 \geq \max\{(\frac{3(2\sqrt{2})^{\alpha}(3\alpha^2-6\alpha+2)\beta\gamma_1 P_{\max}}{(\alpha^2-3\alpha+2)(\gamma_1-\beta)P_{\min}})^{\frac{1}{\alpha-2}} - 2, 4\sigma - 2\}$, packet reception at v_j is always successful as long as there are no other concurrent senders in the square close-in region $a = \frac{\sqrt{2}(2+\Delta_2)r(n)}{4}$, which is contained in disk $D(v_i, (2+\Delta_2)r(n))$. We have thus proven our statement that a transmission from v_i is successful if the other nodes within distance $(2+\Delta_2)r(n)$ from v_i keep silent. This result applies to unicast, broadcast and multicast.

Given the equivalence of physical and protocol interference models established above, it is straightforward to prove $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ in the physical model. The proof follows the same line of reasoning as presented in the protocol model except for replacing Δ with Δ_2. Note that the two conditions needed for the model equivalence, $\rho(n) \geq r(n)$ and the minimum separation $(2+\Delta_2)r(n)$ between concurrent senders in each partition, are satisfied when we choose appropriate values for c_ρ and c_δ.

Step 1, for $\mathbb{I}_{\text{gen}}$. Our proof for the physical model has shown that every successful transmission satisfies SINR $\geq \beta$. Hence, the same transmission must be able to achieve constant rate $B\log_2(1+\beta)$ in the generalized physical model. The bound $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ is then proven trivially by replacing the constant W used in the physical model with the new constant $B\log_2(1+\beta)$.

Step 2. Till now, we have demonstrated $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ for all the models of $\mathbb{I}$, $\mathbb{L}$ and $\mathbb{C}$ when $\rho(n) = c_\rho r(n)$, $\delta(n) = c_\delta r(n)$ and $\xi(n) = c_\xi r(n)$, if we configure constants c_ρ, c_δ and c_ξ properly. Lastly, we generalize this result to obtain the sufficient scaling conditions on $\rho(n)$, $\delta(n)$ and $\xi(n)$.

When $\rho(n) \geq c_\rho r(n)$, there are two possibilities. If $\rho(n) = \Theta(r(n))$, each partition is geographically bounded between inner radius $\rho'(n) = c_\rho r(n)$ and outer radius $\sigma'\rho'(n)$ ($\sigma' = \frac{\sigma\rho(n)}{\rho'(n)}$ is constant). The proof for $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ is the same as before except for replacing σ with σ'. If $\rho(n) = \omega(r(n))$, we break down each partition into smaller partitions that are parameterized by inner radius $\rho'(n) = c_\rho r(n)$ and outer radius $\sigma\rho'(n)$. Since $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$ in every smaller partition, we can merge these partitions back into the original partition and retain $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$.

We note that $\delta(n)$ (resp. $\xi(n)$) describes the minimum (resp. maximum) separation between concurrently scheduled link transmissions. For $\mathcal{S}_l(t)$ with $\delta(n) \leq c_\delta r(n)$ and $\xi(n) \geq c_\xi r(n)$, we can always choose $\delta'(n) = c_\delta r(n) \geq \delta(n)$ and $\xi'(n) = c_\xi r(n) \leq \xi(n)$ to achieve $w_{l,ij}(n) \geq \frac{c}{r^2(n)}$. □

3.5 Necessary Condition for Maximum Capacity

We next prove the necessary condition in Theorem 3.4, which states that $w_{l,ij}(n) = \Omega(w_{g,ij}(n))$ only if $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$. Given our previous result $w_{g,ij}(n) = \Theta(\frac{1}{r^2(n)})$ obtained in Theorem 3.5, it is hence required to prove the necessity of $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$ for $w_{l,ij}(n) = \Omega(\frac{1}{r^2(n)})$. Before proceeding with the proof, we present two lemmas.

***Lemma* 3.3.** *Given sequences $\{f(n)\}$ and $\{g(n)\}$ that satisfy $f(n) \neq \Omega(g(n))$, there exist subsequences $\{f(n_k)\} \subseteq \{f(n)\}$ and $\{g(n_k)\} \subseteq \{g(n)\}$ such that $f(n_k) = o(g(n_k))$, and vice versa.*

Proof. If there are no subsequences that satisfy $f(n_k) = o(g(n_k))$, there must exist a constant $c > 0$ such that $\frac{f(n)}{g(n)} \geq c$ for all n, which implies $f(n) = \Omega(g(n))$. Therefore, $f(n) \neq \Omega(g(n))$ must indicate $f(n_k) = o(g(n_k))$ for some subsequences $\{f(n_k)\}$ and $\{g(n_k)\}$.

If there exist subsequences $f(n_k) = o(g(n_k))$, then we have $\lim_{n_k \to \infty} \frac{f(n_k)}{g(n_k)} = 0$. Hence we cannot find any constant $c > 0$ such that $f(n) \geq cg(n)$ as $n \to \infty$, i.e., $f(n) \neq \Omega(g(n))$. □

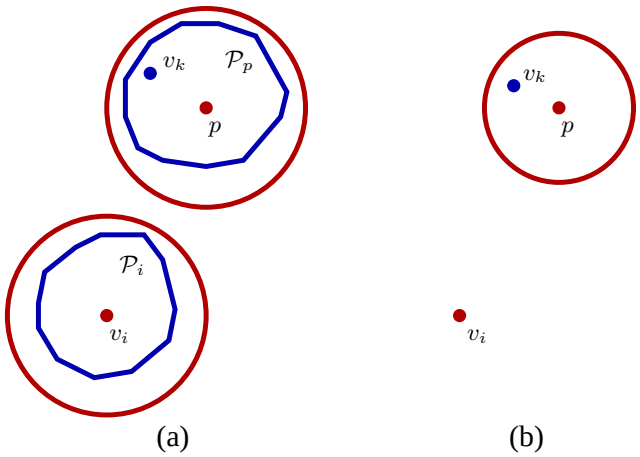


Figure 3.7: Proof of Theorem 3.7 in the protocol interference model. Case (a) proves $\rho(n) = \Omega(r(n))$, in which we define disks $D(v_i, 2\sigma\rho(n))$ and $D(p, 2\sigma\rho(n))$. Case (b) proves $\xi(n) = \Omega(r(n))$, in which we define disk $D(p, \xi(n))$. In both cases, $|\overline{v_i p}| = \frac{\Delta r(n)}{2}$.

***Lemma* 3.4.** *Given sequences $\{f(n)\}$ and $\{g(n)\}$ that satisfy $f(n) \neq O(g(n))$, there exist subsequences $\{f(n_k)\} \subseteq \{f(n)\}$ and $\{g(n_k)\} \subseteq \{g(n)\}$ such that $f(n_k) = \omega(g(n_k))$, and vice versa.*

Proof. This statement is obviously true according to Lemma 3.3, since $f(n) \neq O(g(n))$ means $g(n) \neq \Omega(f(n))$ and $f(n_k) = \omega(g(n_k))$ means $g(n_k) = o(f(n_k))$. □

We then have the next theorem for the necessity of conditions on $\rho(n)$, $\delta(n)$ and $\xi(n)$ to obtain $w_{l,ij}(n) = \Omega(\frac{1}{r^2(n)})$.

***Theorem* 3.7.** *In large-scale wireless networks, $\forall\mathbb{I}, \forall\mathbb{L}, \forall\mathbb{C}$, a localized link scheduling algorithm $\mathcal{S}_l(t)$ achieves $w_{l,ij}(n) = \Omega(\frac{1}{r^2(n)})$ for any link l_{ij} only if $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$.*

Proof. We use contradiction, i.e., $w_{l,ij}(n) \neq \Omega(\frac{1}{r^2(n)})$ if $\rho(n) \neq \Omega(r(n))$ or $\delta(n) \neq O(r(n))$ or $\xi(n) \neq \Omega(r(n))$. The three interference models are discussed separately. As no assumption is made on node locations and communication types, the proof applies to all the models of $\mathbb{L}$ and $\mathbb{C}$.

For $\mathbb{I}_{\text{prot}}$, if $\rho(n) \neq \Omega(r(n))$, Lemma 3.3 states that there exists $\rho(n_k) = o(r(n_k))$. We choose a point p such that $|\overline{v_i p}| = \frac{\Delta r(n)}{2}$ and define disks $D(v_i, 2\sigma\rho(n))$ and $D(p, 2\sigma\rho(n))$, as shown in Fig. 3.7(a). Since the partition diameter does not exceed $2\sigma\rho(n)$, the partition where v_i resides, denoted as $\mathcal{P}_i$, must be contained in $D(v_i, 2\sigma\rho(n))$ and the partition where

p is located, denoted as $\mathcal{P}_p$, must be contained in $D(p, 2\sigma\rho(n))$. Given $\rho(n_k) = o(r(n_k))$, we obtain $\frac{2\sigma\rho(n_k)}{|\overline{v_i p}|} = \frac{2\sigma\rho(n_k)}{\Delta r(n_k)/2} \to 0$ as $n_k \to \infty$. Thus, $\exists n^*$ such that $\forall n_k > n^*$, $\frac{2\sigma\rho(n_k)}{|\overline{v_i p}|} < \frac{1}{2}$, showing that $D(v_i, 2\sigma\rho(n_k))$ and $D(p, 2\sigma\rho(n_k))$ do not overlap and that $\mathcal{P}_i$ and $\mathcal{P}_p$ are separated. Since there is at least one sender v_k in $\mathcal{P}_p$ and $|\overline{v_i v_k}| < \Delta r(n_k)$, collision must occur between v_i and v_k. Consequently, $w_{l,ij}(n_k) = 0$ when $n_k > n^*$. In other words, $w_{l,ij}(n_k) = o(\frac{1}{r^2(n_k)})$ or equivalently $w_{l,ij}(n) \neq \Omega(\frac{1}{r^2(n)})$.

For $\mathbb{I}_{\text{prot}}$, if $\delta(n) \neq O(r(n))$, by Lemma 3.4, there exists $\delta(n_k) = \omega(r(n_k))$. As node density is 1 and a partition cannot schedule multiple concurrent senders in any square region with edge $\frac{\delta(n)}{\sqrt{2}}$, the chance for v_i to be scheduled is at most $\frac{2}{\delta^2(n)}$. The time-average transmission rate of l_{ij} is thus bounded by $w_{l,ij}(n) \leq \frac{2\eta_{ij}W}{\delta^2(n)}$, which gives $\frac{w_{l,ij}(n_k)}{1/r^2(n_k)} \leq \frac{2\eta_{ij}Wr^2(n_k)}{\delta^2(n_k)} \to 0$ as $n_k \to \infty$. Hence, $w_{l,ij}(n_k) = o(\frac{1}{r^2(n_k)})$ and $w_{l,ij}(n) \neq \Omega(\frac{1}{r^2(n)})$.

For $\mathbb{I}_{\text{prot}}$, if $\xi(n) \neq \Omega(r(n))$, by Lemma 3.3, there exists $\xi(n_k) = o(r(n_k))$. We choose a point p such that $|\overline{v_i p}| = \frac{\Delta r(n)}{2}$ and define disk $D(p, \xi(n))$, as shown in Fig. 3.7(b). Because $\frac{\xi(n_k)}{|\overline{v_i p}|} = \frac{\xi(n_k)}{\Delta r(n_k)/2} \to 0$ as $n_k \to \infty$, there must exist n^* such that $\forall n_k > n^*$, $\frac{\xi(n_k)}{|\overline{v_i p}|} < 1$ and v_i is outside $D(p, \xi(n_k))$. Since there is at least one sender v_k in $D(p, \xi(n_k))$ and $|\overline{v_i v_k}| < \Delta r(n_k)$, v_k collides with v_i. Therefore, $w_{l,ij}(n_k) = 0$ when $n_k > n^*$. In other words, $w_{l,ij}(n_k) = o(\frac{1}{r^2(n_k)})$ and $w_{l,ij}(n) \neq \Omega(\frac{1}{r^2(n)})$.

For $\mathbb{I}_{\text{phy}}$, from Eq. (3.3) we have $\frac{P_{\max}}{|X_i - X_j|^\alpha} \geq \beta \frac{P_{\min}}{|X_k - X_j|^\alpha}$ for any node v_k that transmits concurrently with v_i, which leads to $|X_k - X_j| \geq \left(\frac{\beta P_{\min}}{P_{\max}}\right)^{\frac{1}{\alpha}} |X_i - X_j| \geq \left(\frac{\beta P_{\min}}{P_{\max}}\right)^{\frac{1}{\alpha}} r_0(n) = \left(\frac{\beta P_{\min}}{P_{\max}}\right)^{\frac{1}{\alpha}} \varepsilon r(n)$. Hence, when a transmission from v_i to v_j is successful, no senders other than v_i can be scheduled within distance $\left(\frac{\beta P_{\min}}{P_{\max}}\right)^{\frac{1}{\alpha}} \varepsilon r(n)$ from v_j. Along the same line of reasoning as in the protocol model, we prove the necessity of $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$ by replacing v_i with v_j and $\Delta r(n)$ with $\left(\frac{\beta P_{\min}}{P_{\max}}\right)^{\frac{1}{\alpha}} \varepsilon r(n)$.

For $\mathbb{I}_{\text{gen}}$, we first consider the case $\rho(n) \neq \Omega(r(n))$, which implies $\rho(n_k) = o(r(n_k))$. Given a transmission from v_i to v_j, we know that there must exist a node v_k that is transmitting concurrently within distance $4\sigma\rho(n)$ from v_j in a nearby partition, because the partition diameter does not exceed $2\sigma\rho(n)$. Therefore, the SINR at v_j is upper bounded by $\frac{P_{\max}(\varepsilon r(n))^{-\alpha}}{P_{\min}(4\sigma\rho(n))^{-\alpha}}$ and $w_{l,ij}(n_k) \leq \frac{2\eta_{ij}B\log_2(1+\text{SINR})}{(2+\Delta_2)^2 r^2(n_k)} \to \frac{2\eta_{ij}BP_{\max}(\varepsilon r(n_k))^{-\alpha}}{(\ln 2)(2+\Delta_2)^2 r^2(n_k) P_{\min}(4\sigma\rho(n_k))^{-\alpha}}$. Further, $\frac{w_{l,ij}(n_k)}{1/r^2(n_k)} \leq \frac{2\eta_{ij}BP_{\max}(\varepsilon r(n_k))^{-\alpha}}{(\ln 2)(2+\Delta_2)^2 P_{\min}(4\sigma\rho(n_k))^{-\alpha}} \to 0$ as $n_k \to \infty$, showing $w_{l,ij}(n_k) = o(\frac{1}{r^2(n_k)})$ and $w_{l,ij}(n) \neq \Omega(\frac{1}{r^2(n)})$. Hence, $\rho(n) = \Omega(r(n))$ is necessary for $w_{l,ij}(n) = \Omega(\frac{1}{r^2(n)})$. The necessity of condition $\xi(n) = \Omega(r(n))$ is proven similarly. Lastly, if $\delta(n) \neq O(r(n))$, then from Eq. (3.5) we have $\frac{w_{l,ij}(n_k)}{1/r^2(n_k)} \leq \frac{2\eta_{ij}B\log_2(1+\gamma_2\varepsilon^{-\alpha})r^2(n_k)}{\delta^2(n_k)} \to 0$, showing $w_{l,ij}(n_k) = o(\frac{1}{r^2(n_k)})$

and $w_{l,ij}(n) \neq \Omega(\frac{1}{r^2(n)})$. Hence $\delta(n) = O(r(n))$ is also a necessary condition for $w_{l,ij}(n) = \Omega(\frac{1}{r^2(n)})$. □

3.6 Design of Localized Scheduling Scheme

Till now, we have proven Theorem 3.1, the main result of this chapter. The capacity relation $\lambda_l(n) = \Theta(\lambda_g(n))$ implies that the impact of cross-partition collisions is bounded when using localized scheduling such that any failed transmission can eventually get through after a constant number of retries, thus allowing localized scheduling to achieve the same order of capacity as global scheduling. Although our study has focused on the limiting case $n \to \infty$, the result is applicable to finite network sizes too. As long as the parameters $\rho(n)$, $\delta(n)$ and $\xi(n)$ are chosen to the scale of $r(n)$, the network capacity is some constant fraction of that achieved by the ideal global scheduling strategy.

Next, we apply our analytical result to design a localized scheduling scheme to serve as our solution for practically maximizing the capacity of large wireless networks. Our scheme consists of two components, the network partition protocol and the link scheduling algorithm. As mentioned before, the partitions can be formed by designating a set of nodes as schedulers which coordinate the transmissions within their respective neighborhoods. In order to satisfy our requirement on the partition sizes, the chosen schedulers may need to be placed at desired locations. However, it may not be possible to designate the location for each scheduler when the network is large. We therefore propose a distributed protocol that automatically generates network partitions conforming to our requirement.

3.6.1 Distributed Partition Protocol

Our analysis shows that the size of each partition should be at least proportional to the critical transmission radius $r(n)$. As we want to maximize the network capacity and minimize the scheduling overhead simultaneously, we generate the smallest acceptable partitions to minimize the protocol overhead. Specifically, our targeted partition areas should be bounded between an inner disk of radius $\kappa_1 r(n)$ and an outer disk of radius $\kappa_2 r(n)$, where $\kappa_2 > \kappa_1 > 0$ are constants. By utilizing Voronoi Tessellation [56], our task is in fact to determine the subset of nodes that will serve as the schedulers. The non-scheduler nodes simply join the respective nearest schedulers to form partitions.

The basic idea of our distributed partition protocol is the scheduler determination via random competition. Each node in the network volunteers to become a scheduler with some self-recommendation probability. If two nodes that are close to each other claim to be schedulers simultaneously, one gives up the competition such that the generated partitions will conform to our size requirement. The partition process finishes when every node belongs to a partition. Note that there is a tradeoff between the partition overhead and the convergence time that is controlled by the self-recommendation probability. For a higher probability, larger overhead and faster convergence are expected. We configure this probability to be inversely proportional to the node density, which ensures that both the overhead and the convergence are within acceptable bounds.

We present our distributed partition protocol in Fig. 3.8. This protocol runs in every node independently until the node has determined to become a scheduler or it has chosen a nearby scheduler to join. According to the protocol, the undecided nodes first exchange their identification and location information. Based on the received messages, a node, for example v_i, figures out the set of undecided neighbors and their locations. From this node set, v_i determines the smallest subset in which the nodes fall into a disk that has radius $\frac{r(n)}{2}$ and is arbitrarily located around v_i. We denote the cardinality of this smallest subset as N_i. Then v_i volunteers to become a scheduler with probability $\frac{1}{aN_i}$, where a is a positive constant. If v_i decides to be a scheduler, it chooses a random backoff timer T_i for sending its announcement. In case that v_i receives other announcements from nodes within distance $2\kappa_1 r(n)$ before T_i expires, v_i revokes its decision. Otherwise, v_i sends its announcement to distance $\kappa_2 r(n)$ away. If v_i does not decide to serve as a scheduler and not receive the announcements from other nodes, it repeats this protocol from the first step until either itself becomes a scheduler or it joins a nearby scheduler within distance $\kappa_2 r(n)$.

We comment on the node self-recommendation probability $\frac{1}{aN_i}$ and the backoff timer T_i to explain our design rationale. As a scheduler covers a distance up to $\kappa_2 r(n)$, we need roughly only one scheduler in every $\pi\kappa_2^2 r^2(n)$ area. That is, the self-recommendation probability should be inversely proportional to the number of undecided nodes in this area. However, counting the number of undecided nodes in multihop will involve extra overhead. We hence require a node to count the number of undecided nodes in its one-hop neighborhood only and use the constant a as an approximate compensation for counting less nodes. In the protocol we have used the count N_i in an even smaller area $\frac{\pi}{4}r^2(n)$ in order to facilitate the proof of the protocol convergence property, which will be explained in details

Distributed Partition Protocol: DPP	
1	broadcast message $M_i = \{v_i, X_i\}$
2	receive messages $\mathcal{M}_i = \{M_{j_1}, \cdots, M_{j_k}\}$
3	determine the node set $\mathcal{N}_i = \{v_{j_1}, \cdots, v_{j_k}\}$
4	determine $N_i = \min_{X \in D(X_i, \frac{r(n)}{2})} \lvert\mathcal{N}_{i,X}\rvert$, where the node subset $\mathcal{N}_{i,X} = \{v_j : \lvert X - X_j\rvert \leq \frac{r(n)}{2}\}$
5	volunteer to be scheduler with probability $\frac{1}{aN_i}$
6	choose a backoff timer $T_i \in \{0, \varsigma, \cdots, m\varsigma\}$
7	**if** a scheduler announcement is received within distance $2\kappa_1 r(n)$ before T_i expires **then**
8	cancel own scheduler status
9	**else** announce the scheduler status to distance $\kappa_2 r(n)$ at the expiry of T_i
10	**if** self is not a scheduler and other scheduler announcements have not been received **then**
11	repeat from step 1
12	**else** this process ends

Figure 3.8: The process of determining the role of node v_i.

later. The backoff timer T_i is designed to reduce the chance of scheduler confliction. When two nodes are close to each other and volunteer to be schedulers at the same time, random backoff helps resolve the confliction. The interval ς in the protocol is assumed to be the maximum message transmission time between such two nodes. In the unlucky case that two nodes in competition choose the same backoff timer, we require the node with a lower identification number become the scheduler.

It can be proven that the distributed partition protocol designed above generates partitions that satisfy our size requirement. In addition, the protocol converges within a constant number of iterations and the partition overhead per node is bounded by constant. These properties are proven below.

***Property* 3.1.** *Using DPP, each partition is bounded between an inner disk with radius $\kappa_1 r(n)$ and an outer disk with radius $\kappa_2 r(n)$.*

Proof. All we need to show is that the distance $|X_i - X_k|$ from a scheduler v_i to an arbitrary node v_k on the partition boundary is bounded between $\kappa_1 r(n)$ and $\kappa_2 r(n)$. Obviously, $|X_i - X_k| \geq \min_{\{e\}} |X_i - X_{i,e}|$, where $\{e\}$ is the set of edges delimiting the partition boundary and $X_{i,e}$ is the location of v_i's projection on edge e. By the property of Voronoi Tessellation, $|X_i - X_{i,e}| = \frac{1}{2}|X_i - X_j| \geq \kappa_1 r(n)$, where v_j is the neighboring scheduler whose partition

shares edge e with v_i and $|X_i - X_j| \geq 2\kappa_1 r(n)$ according to the protocol. Hence, we have $|X_i - X_k| \geq \kappa_1 r(n)$. On the other hand, since v_i covers other non-scheduler nodes up to a maximum distance of $\kappa_2 r(n)$, its partition must be bounded within an outer disk with radius $\kappa_2 r(n)$. □

***Property* 3.2.** *Using DPP, each node is expected to determine its role within constant number of iterations.*

Proof. The role of a node is determined if itself becomes a scheduler or it joins a scheduler within distance $\kappa_2 r(n)$. We next bound the probability p for a node v_i to determine its role in one iteration of the protocol. We observe that v_i is determined as long as one node volunteers to become a scheduler within distance $(\kappa_2 - 2\kappa_1)r(n)$, including v_i itself. It does not matter if this volunteering node loses competition to another node, because the winning node must become a scheduler within distance $\kappa_2 r(n)$ from v_i. Let $\mathcal{V} = \{v_i, v_{j_1}, \cdots, v_{j_k}\}$ denote the set of undecided nodes within distance $(\kappa_2 - 2\kappa_1)r(n)$ from v_i. We bound p as

$$p \geq 1 - \prod_{v_x \in \mathcal{V}} \Big(1 - \frac{1}{aN_x}\Big). \tag{3.22}$$

According to the protocol, N_x is the minimum number of undecided nodes in a disk of radius $\frac{r(n)}{2}$ around v_x. It is not difficult to find $N_x \leq |\mathcal{V}| = k + 1$. Substituting $k + 1$ for N_x in Eq. (3.22), we have

$$p \geq 1 - \Big(1 - \frac{1}{a(k+1)}\Big)^{k+1}. \tag{3.23}$$

Since function $f(x) = (1 - \frac{1}{x})^x$ $(x \geq 1)$ is an increasing function that converges to $\frac{1}{e}$ as $x \to \infty$, we further have

$$p \geq 1 - e^{-\frac{1}{a}}. \tag{3.24}$$

Hence, the expected number of iterations for v_i to determine its role is upper bounded as $\frac{1}{p} \leq \frac{1}{1-e^{-\frac{1}{a}}}$, which is a constant. □

***Property* 3.3.** *Using DPP, the partition overhead is constant per node on average.*

Proof. The protocol overhead consists of two parts. First, when a node has not decided its role, it broadcasts its status to the neighbors. Second, a scheduler node announces its status to distance $\kappa_2 r(n)$ away. In the first case, the node sends one status message in each iteration of the protocol until its status is determined. According to Property 3.2, the expected number of status messages sent by a node does not exceed $\frac{1}{1-e^{-\frac{1}{a}}}$. In the

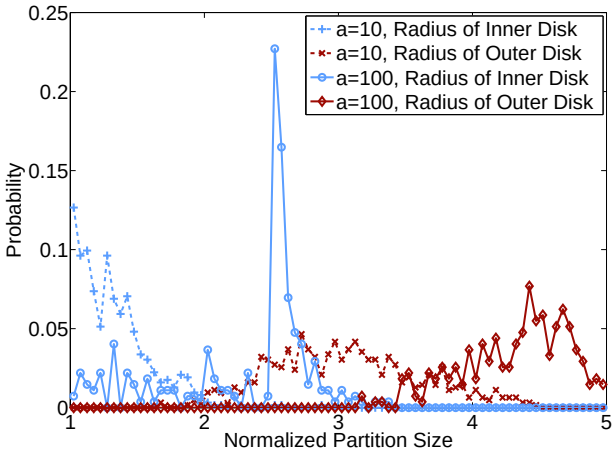


Figure 3.9: Probability distribution of the partition sizes normalized against $r(n)$.

second case, a node helps forward the scheduler announcement each time a node becomes a scheduler within distance $\kappa_2 r(n)$. By the protocol, one node volunteers to be a scheduler in every area $\frac{\pi a}{4} r^2(n)$ on average. Hence, there are $\frac{4\kappa_2^2}{a}$ schedulers on average around every node in the network and each node forwards $\frac{4\kappa_2^2}{a}$ number of scheduler announcements. Putting the overhead together, the total number of messages sent or forwarded by each node does not exceed $\frac{1}{1-e^{-\frac{1}{a}}} + \frac{4\kappa_2^2}{a}$ on average, which is a constant. □

These properties of the partition protocol are also verified by simulations. In our simulations, we generate $n = 10^5$ nodes with random locations and set the node transmission radius as $r(n) = \sqrt{\ln n} = 3.39$. The partition protocol constants are set as $\kappa_1 = 1$, $\kappa_2 = 5$, $a = 10$ or $a = 100$, and $m = 10$. We run the protocol once and plot the distribution of partition sizes in Fig. 3.9. We observe that the parameter a determines the partition sizes. When a increases, the distribution of the partition sizes shifts toward large values. The schedulers scatter sparsely when a is large because each node volunteers to become a scheduler with reduced probability. The generated partitions are hence larger than those with small a. We repeat the simulation for 500 times and plot in Fig. 3.10 the minimum radius of the inner disk and the maximum radius of the outer disk for all the partitions generated at the end of each simulation run. It is observed that the partition sizes in all the simulation runs are bounded between $r(n)$ ($\kappa_1 = 1$) and $5r(n)$ ($\kappa_2 = 5$). Thus, the partition protocol is able to generate satisfactorily sized partitions by configuring κ_1 and

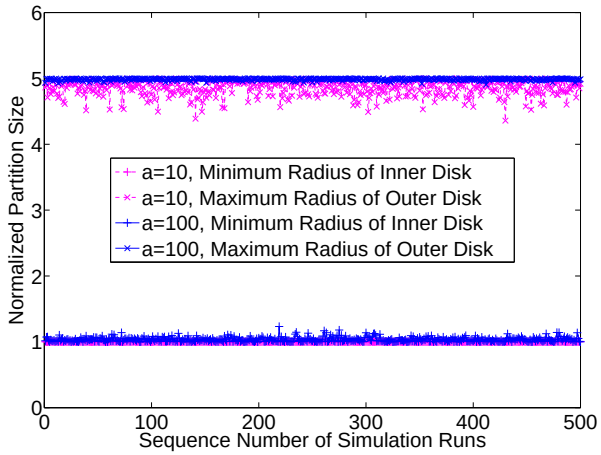


Figure 3.10: Upper and lower bounds on the partition sizes normalized against $r(n)$.

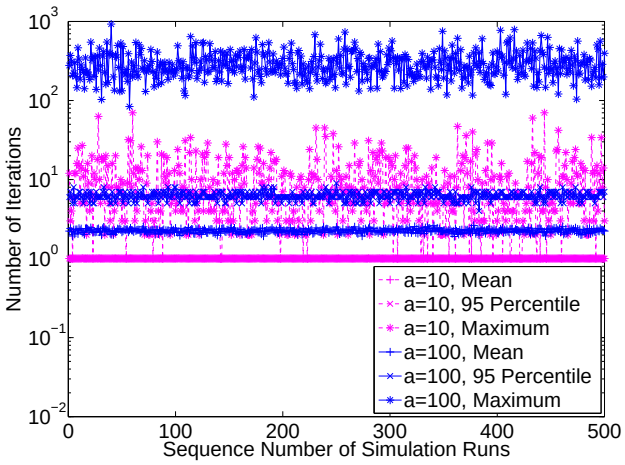


Figure 3.11: Protocol convergence.

κ_2. We count the number of iterations needed for each node to determine its role and plot the mean, 95% percentile and maximum values for each simulation run in Fig. 3.11. Our analysis shows that the average convergence time is upper bounded by $\frac{1}{1-e^{-\frac{1}{a}}}$, which equals 10.5 and 100.5 when $a = 10$ and $a = 100$ respectively. The simulation result of the mean iteration numbers agrees well with our analysis. We also observe from Fig. 3.11 that a few nodes may have a long convergence time, but 95% majority converges within a time frame comparable to the mean iteration numbers. Similarly, we plot the mean, 95% percentile and

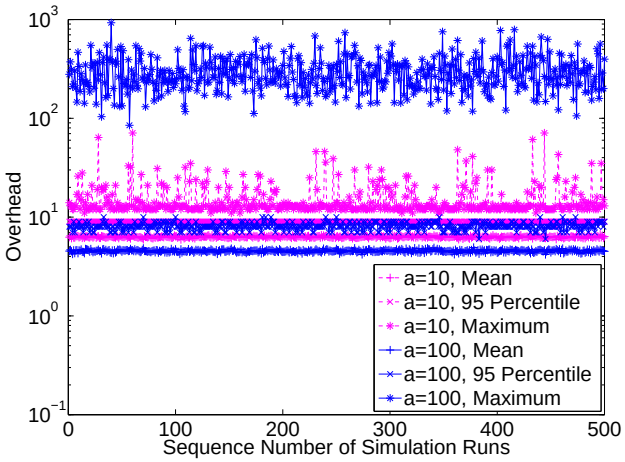


Figure 3.12: Protocol overhead.

Localized Scheduling Algorithm: LSA

Input: a set of links $\mathcal{L}$ requesting transmission
Output: a schedule $\mathcal{S}_l(t)$ satisfying **S1**, **S2** and **S3**

$\mathcal{S}_l(t) \leftarrow \emptyset$
while $\mathcal{L} \neq \emptyset$
    select a link l_{ij} randomly from $\mathcal{L}$
    $\mathcal{S}_l(t) \leftarrow \mathcal{S}_l(t) \cup \{l_{ij}\}$
    $\mathcal{L} \leftarrow \mathcal{L} \setminus \{l_{ij}\}$
    $\mathcal{L} \leftarrow \mathcal{L} \setminus \{l_{i'j'} : |X_i - X_{i'}| < \delta(n)\}$
return $\mathcal{S}_l(t)$

Figure 3.13: A localized scheduling algorithm that runs in each partition and in each time slot t.

maximum protocol overhead for each simulation run in Fig. 3.12. According to analysis, the average overhead per node does not exceed $\frac{1}{1-e^{-\frac{1}{a}}} + \frac{4\kappa_2^2}{a}$. With $a = 10$ and $a = 100$, this upper bound equals 20.5 and 101.5 respectively. The simulation result of the mean overhead verifies our analysis.

3.6.2 Localized Scheduling Algorithm

Given the partitions determined by the schedulers, we next present a localized scheduling algorithm as in Fig. 3.13, which runs at the scheduler in each partition. The scheduler collects the topology information inside its partition and generates a collision-free schedule in each time slot. The algorithm is parameterized as: $\sigma = 10$, $\rho(n) = r(n)$,

$\delta(n) = 3(1+\Delta)r(n)$ and $\xi(n) = 4(1+\Delta)r(n)$ (in $\mathbb{I}_{\text{prot}}$) or $\delta(n) = 3(1+\Delta_2)r(n)$ and $\xi(n) = 4(1+\Delta_2)r(n)$ (in $\mathbb{I}_{\text{phy}}$ and $\mathbb{I}_{\text{gen}}$). If there are broad/multi-cast communications, the links in the same group of broad/multi-cast branches are taken as one link for the scheduling purpose. The proposed scheduling algorithm has the following properties.

***Property* 3.4.** *Using LSA,* $\lambda_l(n) = \Theta(\lambda_g(n))$.

Proof. It is clear that the configuration of LSA satisfies $\rho(n) = \Omega(r(n))$, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$. Our task is then to prove that LSA satisfies **S1**, **S2** and **S3** so that we can use Theorem 3.1 to prove $\lambda_l(n) = \Theta(\lambda_g(n))$.

Requirement **S1** is obviously met by setting $\rho(n) = r(n)$ and $\sigma = 10$. Requirement **S2** is satisfied via line 6 of LSA, which excludes any link from being scheduled if its distance from a scheduled link is less than $\delta(n)$. Therefore, a minimum space $\delta(n)$ is guaranteed between any two active links in the same schedule. Lastly, we show that there exists at least one active link within radius $\xi(n)$ from any arbitrary location in a partition, to satisfy **S3**. We first consider the interference model $\mathbb{I}_{\text{prot}}$. Given an arbitrary point p in a partition, because $\xi(n) - \delta(n) = (1+\Delta)r(n)$, the transmitting links scheduled outside $D(p, \xi(n))$ do not suppress transmissions in the disk $D(p, (1+\Delta)r(n))$. According to LSA, if there exist any links in $D(p, (1+\Delta)r(n))$, at least one of them is scheduled. We next show that such links do exist. For random node locations, as $n \to \infty$, there exist nodes in $D(p, \Delta r(n))$ almost surely. For arbitrary node locations, given any $r(n) \geq \frac{\sqrt{2}}{\Delta}$, there is at least one node in $D(p, \Delta r(n))$. Moreover, as the length of any link does not exceed the critical transmission radius $r(n)$, a node in $D(p, \Delta r(n))$ forms at least one link with other nodes in $D(p, (1+\Delta)r(n))$. This proves existence of at least one active link in $D(p, \xi(n))$. The proof is the same for interference models $\mathbb{I}_{\text{phy}}$ and $\mathbb{I}_{\text{gen}}$ by replacing Δ with Δ_2. □

***Property* 3.5.** *LSA finishes in constant number of steps.*

Proof. As LSA selects the subset of transmitting links randomly, we model the number of steps as a random variable Z_1 and denote its maximum as $\hat{Z}_1$. Besides, we consider two random experiments of covering given regions with uniform disks of radius $\delta(n)$. The goal is to cover the specified regions seamlessly while keeping a minimum separation $\delta(n)$ between the centers of neighbor disks. In the first experiment we cover a partition. In the second experiment we cover a square region $2\sigma\rho(n) \times 2\sigma\rho(n)$ that is co-centric with the partition. We define random variables Z_2 and Z_3 as the number of disks used in the first

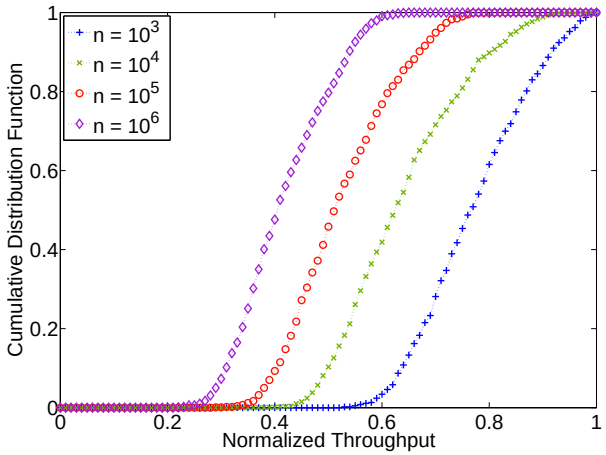


Figure 3.14: Scaling of partition size, $\rho(n) = 1$.

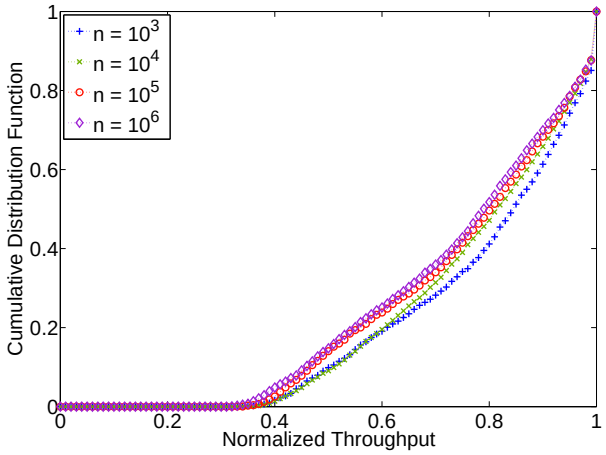


Figure 3.15: Scaling of partition size, $\rho(n) = r(n)$.

and the second experiments, and denote their maxima as $\hat{Z}_2$ and $\hat{Z}_3$ respectively. It is not difficult to find the following relations: $Z_1 \leq \hat{Z}_1 \leq \hat{Z}_2 \leq \hat{Z}_3$. Furthermore, the number of disks needed to cover the square region never exceeds $\frac{(2\sigma\rho(n))^2}{(\pi/4)(\delta(n)/2)^2} = \frac{64\sigma^2}{9\pi(1+\Delta)^2}$ (in $\mathbb{I}_{\text{prot}}$) or $\frac{64\sigma^2}{9\pi(1+\Delta_2)^2}$ (in $\mathbb{I}_{\text{phy}}$ and $\mathbb{I}_{\text{gen}}$). Therefore, $Z_1 \leq \hat{Z}_3 \leq \frac{64\sigma^2}{9\pi(1+\Delta)^2}$ or $Z_1 \leq \hat{Z}_3 \leq \frac{64\sigma^2}{9\pi(1+\Delta_2)^2}$, which proves that LSA finishes in constant number of steps regardless of network size n. □

We have simulated the performance of LSA in random networks to verify its optimality in capacity scaling. We increase the network size n from 10^3 to 10^6. The traffic

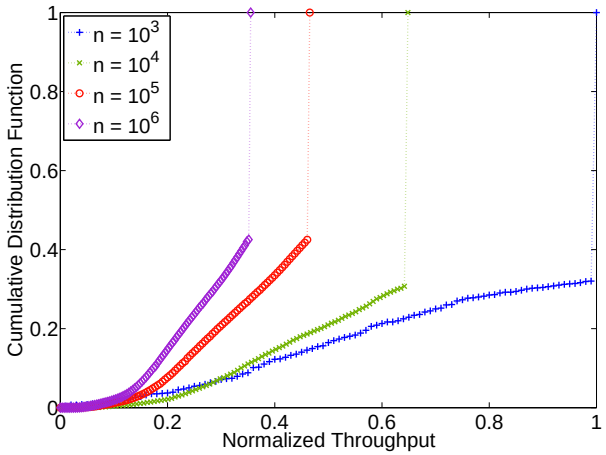


Figure 3.16: Scaling of scheduling density, $\delta(n) = r^{\frac{3}{2}}(n)$.

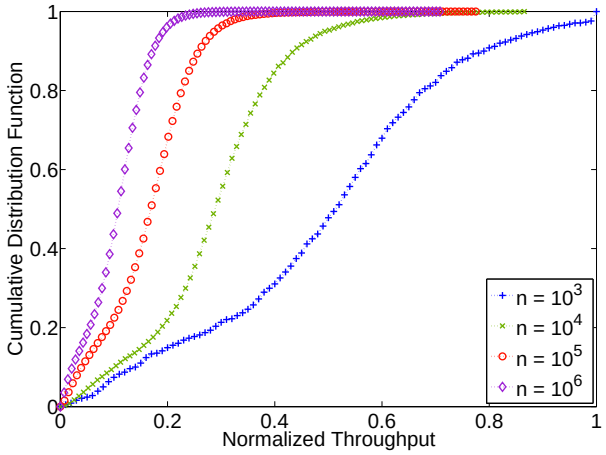


Figure 3.17: Scaling of scheduling density, $\xi(n) = r^{\frac{1}{2}}(n)$.

in the network is a random mixture of unicast, multicast and broadcast. We first choose the partition size from a range of values on different orders of $r(n)$ and plot the cumulative distribution function of the achieved throughput in Fig. 3.14 and Fig. 3.15, where the throughput is normalized against that of $n = 10^3$. It is observed in Fig. 3.14 that the curve of cumulative distribution function moves left-ward as n increases, indicating a trend of diminishing throughput when the partition size is on an order lower than $r(n)$. In contrast, the curve of cumulative distribution function stays static when n increases in Fig. 3.15,

where the partition size scales on the same order as $r(n)$. Hence the simulation results verify that LSA achieves the optimal capacity when $\rho(n) = \Omega(r(n))$. Similarly, we plot in Fig. 3.16 and Fig. 3.17 the cumulative distribution function of the achieved throughput when the scheduling density is not on the order of $r(n)$. We see that the achieved throughput diminishes as n increases when $\delta(n) = \omega(r(n))$ or $\xi(n) = o(r(n))$. Hence, $\delta(n) = O(r(n))$ and $\xi(n) = \Omega(r(n))$ are required in order to achieve the optimal capacity scaling.

3.7 Relax of Time Synchronization

Till now, we have assumed a synchronized clock across the entire network in our analysis and design. Next, we relax this assumption by showing that our proposed partition protocol and scheduling algorithm still work as expected if each node has its own clock that is not synchronized with others, as long as the clock difference displays a geographical dependence. In practice, as synchronizing clocks that are geographically close is comparatively easier than that for nodes located far apart, we assume that the clock difference $\tau(v_i, v_j)$ between two nodes v_i and v_j is a monotonically increasing function of their distance, i.e., $\tau(v_i, v_j) = \tau(\frac{|X_i - X_j|}{r(n)})$ and $\tau(v_i, v_j) < \tau(v_i, v_k)$ if $|X_i - X_j| < |X_i - X_k|$. We next discuss the effect of clock non-synchronization in the operation of our localized scheduling scheme.

3.7.1 Clock Non-Synchronization in Partitioning

According to the partition protocol, each node determines its role in a number of iterations. One iteration includes two stages: status exchange and scheduler announcement. A node receives messages from other nodes in both stages to make its own role decision. Due to clock difference, messages may be received at times different from their expectations. However, we can bound the time offset by utilizing the geographical dependence of clock difference and prove that the partition protocol still works as expected after a slight modification.

In the stage of status exchange, each node, for example v_i, broadcasts its own status and receives the status messages from neighbors. As the distance from v_i to each neighbor does not exceed $r(n)$, the clock difference between v_i and each neighbor is upper bounded by $\tau(1)$. By denoting ϱ as the time window needed to receive all the messages from neighbors when their clocks are synchronized, we extend ϱ by adding an extra time period of $\tau(1)$ before and after ϱ. The new window $\varrho + 2\tau(1)$ is sufficiently large to receive all the

status messages from neighbors. Similarly, we extend the time window ς to $\varsigma + 2\tau(\kappa_2)$ for each node to receive the scheduler announcement if there is a scheduler nearby, since the distance from v_i to any nearby scheduler does not exceed $\kappa_2 r(n)$. According to the protocol, an iteration consists of one status exchange step and $m+1$ scheduler announcement steps. We hence concatenate these steps by using the extended time windows to mitigate the clock non-synchronization problem. Note that we can merge the appending $\tau(1)$ or $\tau(\kappa_2)$ of each step with the preceding $\tau(1)$ or $\tau(\kappa_2)$ of the immediately successive step. The modified protocol generates the same set of partitions as the original protocol. Since $\tau(1)$ and $\tau(\kappa_2)$ are constants, the protocol convergence property remains the same as before.

3.7.2 Clock Non-Synchronization in Scheduling

After the partitions are formed, the scheduler in each partition is responsible for generating a collision-free transmission schedule in its partition. Though the scheduler is able to avoid transmission collisions by allocating different time slots to the interfering transmissions, we need each node to be cooperative by following the schedule faithfully. When the clocks are not synchronized, collisions may still appear. Specifically, if a node sends packets later than its designated time while another node sends ahead of its schedule, their transmissions may overlap and collision may appear. We are however able to bound the extent of overlap by using the geographical dependence of the clock difference. Since the diameter of each partition does not exceed $2\kappa_2 r(n)$, the clock difference between any two nodes in a partition is upper bounded by $\tau(2\kappa_2)$. Hence, we can extend the length of each data transmission time slot by an additional $\tau(2\kappa_2)$ to mitigate the effect of clock non-synchronization. It is not difficult to find that any two interfering transmissions in the same partition will not overlap after such an extension is made to each time slot. Note that our localized scheduling scheme only requires collision-free transmissions within each individual partition. Collisions due to interference from different partitions are allowed. Therefore, clock non-synchronization does not introduce any scheduling problem across partitions.

3.8 Summary

In this chapter, we have proposed the scheduling partition methodology to achieve the order optimal capacity scaling in large wireless networks. It bridges the gap between the current studies on wireless network capacity and the existing schemes on link transmission

scheduling. The current wireless network capacity studies require a globally collision-free transmission schedule to maximize the network utilization, which however cannot be obtained by applying the existing scheduling schemes due to their excessive complexity in large networks. Our new approach divides a large network into small partitions and schedules transmissions in each partition independently. As a result, the scheduling complexity is reduced significantly. Meanwhile, it achieves the same order of capacity scaling as the theoretically derived capacity bounds. Based on the design principles that we have found, we have proposed a distributed protocol to divide a large network into satisfactory partitions and a localized algorithm to schedule collision-free transmissions in each partition. Analysis and simulation show that our scheduling partition approach provides a practical solution for maximizing the capacity of large wireless networks.

Chapter 4

The Topological Stability of Hierarchical Wireless Networks

Our work in the previous two chapters has addressed the minimum packet delay and maximum network capacity problems in large-scale wireless networks. In this and next chapters, we study the resilience of large wireless networks in presence of node mobility and failure. Specifically, we investigate in this chapter the topological stability of the hierarchical network architecture when nodes are free to move. Although many solutions have been proposed in the literature to construct stable hierarchical architecture in large wireless networks, the maximum stability achievable in the mobile environment is still unknown. We define three metrics to measure the network topological stability: the cluster lifetime, the inter-cluster link lifetime, and the end-to-end path lifetime. We model and analyze the maximum of these lifetimes under the constraint of random node mobility. The analytical results provide the fundamental understanding of the bounds on network topological stability. Inspired by this understanding, we propose a clustering algorithm and a hierarchical routing protocol to maximize the topological stability of mobile wireless networks.

4.1 Motivation and Related Work

By organizing large-scale mobile wireless networks into hierarchical architecture, network resources can be allocated and utilized efficiently, thus improving the network performance. However, node mobility poses a big challenge to the stability of network topology, which in turn incurs negative impact on the network performance. When a node

moves, it may be attached to different clusters at different times, resulting in frequent path rediscovery each time it changes the point of attachment. The inter-cluster connectivity affects path stability too. When an inter-cluster link fails, all the communication paths traversing the broken link have to be replaced. The frequent communication interruptions significantly jeopardize network performance, causing low throughput, long delay and high overhead. Ideally, the topological stability of the clusters and their connections should be maximized in order to optimize the network performance.

We define three metrics to quantify the hierarchical network stability: the cluster lifetime, the inter-cluster link lifetime, and the end-to-end path lifetime. These three correlated metrics measure different stability aspects of the hierarchical structure. The cluster lifetime indicates how often the nodes change their cluster memberships, the inter-cluster link lifetime assesses how long neighbor clusters remain connected, and the path lifetime evaluates how stable an end-to-end communication path can be. It is obvious that long lifetime implies stable architecture and good communication performance. Hence, the objective of optimizing the network performance translates into maximizing these three stability-indicating lifetimes.

There has been abundance of work done in the literature that constructs hierarchical network architecture [57–66]. Some have considered node mobility in order to establish stable clusters [67–72] and some have proposed strategies to minimize the cluster changes [68, 73, 74]. All these efforts are effective to improve the cluster stability. However, since these schemes are all heuristic based, they do not reveal the longest possible cluster lifetime in mobile environments. A lot of research has also been conducted on the hierarchical routing protocols [73, 75–83], but none of them has provided an in-depth stability analysis of the links and paths established by these routing protocols. Besides, though the link and path lifetime has been investigated in flat networks [84–88], little has been done towards understanding the inter-cluster link lifetime and the cluster-based path lifetime in hierarchical networks. Understanding the maximum lifetimes of the clusters, the inter-cluster links and the end-to-end paths will greatly advance our knowledge on the performance bounds of hierarchical architectures and, in addition, provide valuable guidance for network architecture design.

We study in this chapter the impact of random node mobility on the topological stability of hierarchical networks. Specifically, we investigate the *maximum* of the cluster lifetime, the inter-cluster link lifetime and the path lifetime in a given mobility environment,

through both analysis and simulation. We then apply the stability results to design a new clustering algorithm and a new cluster-based routing protocol that achieve the maximum stability. The proposed algorithm and protocol can be used in large-scale mobile ad hoc networks to optimize the communication quality and the network resource utilization.

4.2 Models and Assumptions

4.2.1 Mobility Model

The node mobility has been studied extensively in the literature and represented by many mobility models [88–97]. Since different mobilities have different degrees of impact on network topology, we specify the following flexible node mobility model that can be tuned easily to simulate various mobility scenarios. It is similar to the Random Walk [89], but with some modifications to represent more realistic moving behaviors. In this model, a node alternates between the moving and the pausing phases. It selects a random direction from $[0, 2\pi)$ and a random speed from $[v_{min}, v_{max}]$ at the beginning of a moving phase, which is the same as in the Random Walk. But unlike the constant movement duration or distance, a node chooses its travel distance that is a random variable distributed uniformly in $[0, d_{max}]$. If the node hits network boundary before having finished its planned travel distance, it bounces back into the network. Upon arrival at the destination, the node pauses for a random time before starting another movement. The pause (stationary) time is exponentially distributed with mean τ_s. By tuning the moving speed and the pause time, we are able to study the network stability with various node mobilities.

4.2.2 Network Model

We assume the network model as follows. The network consists of N nodes, which are uniformly distributed over an area of $l \times l$ square meters initially. Every node moves independently and obeys the mobility model defined above. The mobility model maintains uniform node spacial distribution over time. The radio transmission radius is r meters for every node. When two nodes are within r meters distance from each other, they are able to communicate directly; otherwise direct communication is not possible. Because we focus on node mobility, we ignore the link disruptions due to wireless signal interferences and obstructions. As such, link existence is solely determined by the inter-node distance. The

cluster is defined to be a group of geographically close nodes in which one node acts as the clusterhead and the others are clustermembers. Every clustermember is a neighbor of its clusterhead. Communication paths are established and packets are forwarded through the overlay network composed by the clusterheads.

4.3 Lifetime Analysis of the Hierarchical Architecture

4.3.1 Cluster Lifetime

We measure a cluster's stability by defining the *Cluster Lifetime* which is the duration of its clusterhead staying in the head status without interruption. The cluster lifetime starts when its clusterhead is designated and ends when the clusterhead switches to a clustermember. Clearly, the various clustering algorithms reconstruct clusters per different requirements, resulting in different ending times of a clusterhead. We are interested here in the *longest* cluster lifetime allowed in a given mobility environment. We hence assume that a clusterhead undertakes its role until all of its affiliated clustermembers have moved away. A coming clusterhead does not force an existing valid cluster to reconstruct. The cluster lifetime measured in this way is the *maximum* in a mobile environment.

Since the cluster lifetime is determined by the time when the last clustermember leaves the clusterhead, it is easier to look at a clustermember's membership duration first. The *longest* membership duration takes place when the clustermember is affiliated to its selected clusterhead all the time until they move to r meters apart. It is determined by their neighboring time. We denote the longest membership time as the random variable T_m and discuss it in three cases corresponding to the initial mobility phases of the clusterhead and the clustermember at the time of cluster formation: 1) both nodes are stationary, 2) one node is stationary and the other is moving, and 3) both nodes are moving.

In the first case, for the ease of analysis we first assume that one node is fixed to its location such that it never moves. With this node fixed, the other node pauses and moves in its vicinity. At the end of a pausing phase, if the movable node chooses a destination that is still reachable from the fixed node, the two nodes will stay in contact for another pause time. T_m in this case consists of a sequence of alternated intervals of pauses and movements until the movable node finally chooses a destination out of the reach of the fixed node. Fig. 4.1 describes the interval sequence, in which T_1 and T_2 represent the random durations

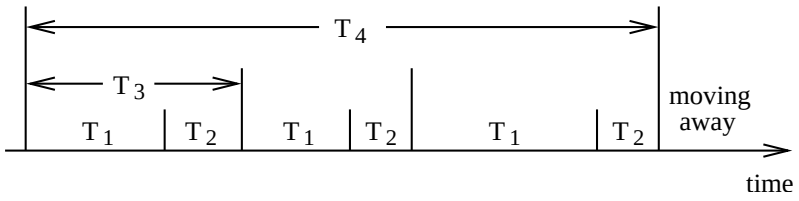


Figure 4.1: The intervals of pauses and movements.

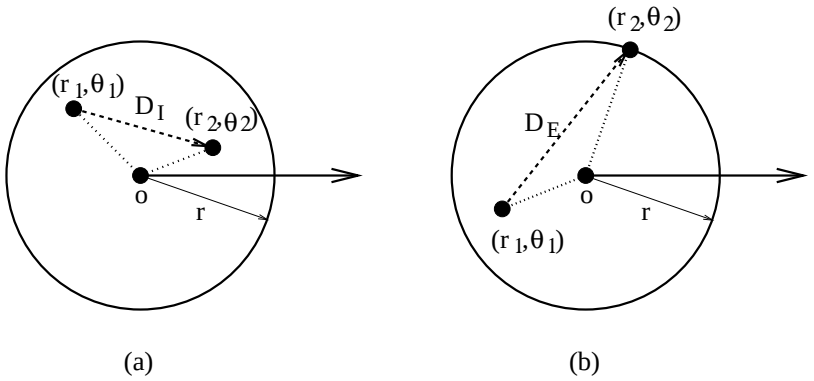


Figure 4.2: The movements inside and beyond the coverage area of another node.

in a pausing phase and in a moving phase respectively. T_1 is exponentially distributed with the mean

$$E(T_1) = \tau_s, \tag{4.1}$$

as specified in the mobility model. The mean of T_2 is determined as

$$E(T_2) = \tau_{1,I} = E(\frac{D_I}{V}) = E(D_I)E(\frac{1}{V}), \tag{4.2}$$

where $\tau_{1,I}$ denotes the mean time of *one* node moving *inside* the transmission area of the fixed node, D_I is the random travel distance *inside* the transmission area, and V is the random speed. We illustrate such a movement in the polar coordinates in Fig. 4.2(a), in which the fixed node is located at the origin. The mean travel distance $E(D_I)$ is computed as

$$\begin{aligned} E(D_I) &= \frac{1}{(\pi r^2)^2} \int_0^{2\pi} \int_0^{r} \int_0^{2\pi} \int_0^{r} d(r_1, \theta_1, r_2, \theta_2) r_1 \, \mathrm{d}r_1 \, \mathrm{d}\theta_1 r_2 \, \mathrm{d}r_2 \, \mathrm{d}\theta_2 \\ &\approx \frac{1}{(\pi r^2)^2} \sum_{\{r_1, \theta_1, r_2, \theta_2\}} d(r_1, \theta_1, r_2, \theta_2) r_1 (\frac{r}{n})(\frac{2\pi}{n}) r_2 (\frac{r}{n})(\frac{2\pi}{n}) \\ &= \frac{4}{n^4 r^2} \sum_{\{r_1, \theta_1, r_2, \theta_2\}} d(r_1, \theta_1, r_2, \theta_2) r_1 r_2, \end{aligned} \tag{4.3}$$

where $d(r_1, \theta_1, r_2, \theta_2) = \sqrt{(r_1 \cos\theta_1 - r_2 \cos\theta_2)^2 + (r_1 \sin\theta_1 - r_2 \sin\theta_2)^2}$. Because there is no closed-form solution for the integral in (4.3), we have used the numerical approximation of dividing the integration range of each variable into n fragments and summing the integration results over all the fragments. The speed V is uniformly distributed and the mean of its inverse is

$$E(\frac{1}{V}) = \int_{v_{min}}^{v_{max}} \frac{1}{v} \cdot \frac{1}{v_{max} - v_{min}} \, \mathrm{d}v = \frac{\ln(v_{max}) - \ln(v_{min})}{v_{max} - v_{min}}. \tag{4.4}$$

We define the sum of T_1 and T_2 as another random variable T_3. For mathematical manageability, the distribution of T_3 is approximated by an exponential distribution. Its mean is

$$E(T_3) = \tau_s + \tau_{1,I}. \tag{4.5}$$

As there is a probability of

$$P_O = 1 - \frac{\pi r^2}{\pi d_{max}^2} = \frac{d_{max}^2 - r^2}{d_{max}^2} \tag{4.6}$$

that the movable node travels to a destination beyond the reach of the fixed node in each moving phase, the total neighboring time T_4 is an exponentially distributed random variable with the mean

$$E(T_4) = \frac{E(T_3)}{P_O}. \tag{4.7}$$

Then we remove the assumption of one node fixed to consider the independent movements of both nodes. Let us denote the two nodes as A and B, and denote the neighboring time when A is fixed as $T_{4,A}$ and the neighboring time when B is fixed as $T_{4,B}$. We have

$$T_m = \min(T_{4,A}, T_{4,B}) \tag{4.8}$$

and determine its mean as

$$E_{s,s}(T_m) = \frac{E(T_4)}{2} = \frac{\tau_s + \tau_{1,I}}{2P_O}, \tag{4.9}$$

where $E_{s,s}(T_m)$ denotes the mean in the case of both nodes being initially stationary.

In the second case, one node is stationary and the other is moving initially. With probability

$$P_I = 1 - P_O = \frac{r^2}{d_{max}^2}, \tag{4.10}$$

the moving node stops inside the transmission area of the stationary node. The mean of this duration is $\tau_{1,I}$. After the moving node has stopped, the rest neighboring time is exactly the

same as what we have discussed before in the case 1. The mean duration is $E_{s,s}(T_m)$. With probability P_O, the moving node has a destination outside the coverage of the stationary node. Let us denote the time before the two nodes become r meters apart as T_5, its mean is determined by

$$E(T_5) = \tau_{1,E} = E(\frac{D_E}{V}) = E(D_E)E(\frac{1}{V}), \tag{4.11}$$

where $\tau_{1,E}$ denotes the mean time of *one* node moving to the *edge* of the transmission area of the stationary node. The random travel distance to the *edge*, denoted as D_E, is shown in Fig. 4.2(b). Its mean is

$$\begin{aligned} E(D_E) &= \frac{1}{2\pi \cdot \pi r^2} \int_0^{2\pi} \int_0^{2\pi} \int_0^{r} d(r_1, \theta_1, \theta_2)\, r_1 \mathrm{d}r_1 \mathrm{d}\theta_1\, \mathrm{d}\theta_2, \\ &\approx \frac{1}{2\pi \cdot \pi r^2} \sum_{\{r_1,\theta_1,r_2\}} d(r_1, \theta_1, \theta_2)\, r_1 (\frac{r}{n})(\frac{2\pi}{n})(\frac{2\pi}{n}) \\ &= \frac{2}{n^3 r} \sum_{\{r_1,\theta_1,\theta_2\}} d(r_1, \theta_1, \theta_2)\, r_1, \end{aligned} \tag{4.12}$$

where $d(r_1, \theta_1, \theta_2) = \sqrt{(r_1 \cos\theta_1 - r\cos\theta_2)^2 + (r_1 \sin\theta_1 - r\sin\theta_2)^2}$. The integral in (4.12) does not have a closed-form solution, so we have used a numerical approximation similar to (4.3). Summarizing both possibilities, we have the mean membership time in the case of one node being initially stationary and the other being initially moving as

$$E_{s,m}(T_m) = P_I(\tau_{1,I} + E_{s,s}(T_m)) + P_O \tau_{1,E}. \tag{4.13}$$

In the third case, both nodes are moving initially. With probability P_I, they stop within each other's transmission area. Denoting the time to one node stopping as T_6, we have

$$E(T_6) = \tau_{2,I} = E(\frac{D_I}{V_R}) = E(D_I)E(\frac{1}{V_R}), \tag{4.14}$$

where $\tau_{2,I}$ denotes the mean time of *two* nodes moving *inside* each other's transmission area, and V_R is their relative speed. The mean of $\frac{1}{V_R}$ can be obtained as

$$\begin{aligned} E(\frac{1}{V_R}) &= \frac{1}{\pi(v_{max} - v_{min})^2} \int_{v_{min}}^{v_{max}} \int_{v_{min}}^{v_{max}} \int_0^{\pi} \frac{1}{V_R}\, \mathrm{d}\theta \mathrm{d}v_1 \mathrm{d}v_2 \\ &\approx \frac{1}{\pi(v_{max} - v_{min})^2} \sum_{\{\theta,v_1,v_2\}} \frac{1}{V_R}\, (\frac{\pi}{n})(\frac{v_{max} - v_{min}}{n})^2 \\ &= \frac{1}{n^3} \sum_{\{\theta,v_1,v_2\}} \frac{1}{V_R}, \end{aligned} \tag{4.15}$$

where $V_R = \sqrt{v_1^2 + v_2^2 - 2v_1 v_2 \cos\theta}$. We have used in (4.15) the numerical computation for the integral as before. The mean of the random travel distance $E(D_I)$ is obtained from (4.3). After one node has come to stop, the time to the other node stopping is described by T_2 and we have $E(T_2) = \tau_{1,I}$ as before. After both nodes become stationary, the rest of their neighboring time is again the case 1 that we have discussed earlier, which has the mean duration of $E_{s,s}(T_m)$. With probability P_O, the two initially moving nodes will move apart. There are two possibilities of their mobility phases at the time of being apart: both are moving or one is moving and the other is stationary. Let us denote the respective probabilities as γ and $1-\gamma$. In the former case, we denote the random neighboring time as T_7 and obtain its mean as

$$E(T_7) = \tau_{2,E} = E(\frac{D_E}{V_R}) = E(D_E)E(\frac{1}{V_R}), \tag{4.16}$$

where $\tau_{2,E}$ denotes the mean time of *two* nodes moving to the *edge* of each other's transmission area. $E(D_E)$ and $E(\frac{1}{V_R})$ are obtained from (4.12) and (4.15) respectively. In the latter case, one node stops inside the transmission area of the other node first and then the other node continues to move away. The time to one node stopping is described by the random variable T_6 with the mean $\tau_{2,I}$ and the time to the other node moving away is described by T_5 with the mean $\tau_{1,E}$, as we have discussed before. γ is determined as follows. We first determine the probability that a node stops before they are apart as

$$\begin{aligned} Q &\approx P\Big\{\frac{W \cdot X}{V} < \tau_{2,E}\Big\} \\ &= \int\int\int_{\frac{w\cdot x}{v} < \tau_{2,E}} \frac{1}{d_{max}(v_{max} - v_{min})} \mathrm{d}w\mathrm{d}x\mathrm{d}v \\ &\approx \frac{1}{n^3} \sum_{\{w,x,v\}} 1_{\{\frac{w\cdot x}{v} < \tau_{2,E}\}}(w,x,v) \end{aligned} \tag{4.17}$$

where W is uniformly distributed in $[0,1]$, X is the random travel distance, $W \cdot X$ is the residual travel distance at the time of cluster formation, and $1_{\{\frac{w\cdot x}{v} < \tau_{2,E}\}}(w,x,v)$ is an indicator function. Q is computed by numerical approximation. Then the probability of both nodes moving at the time they are apart is $(1-Q)^2$ and the probability of one moving and one stationary is $2Q(1-Q)$. By normalization,

$$\gamma = \frac{(1-Q)^2}{(1-Q)^2 + 2Q(1-Q)}, \qquad 1-\gamma = \frac{2Q(1-Q)}{(1-Q)^2 + 2Q(1-Q)}. \tag{4.18}$$

Considering both probabilities of P_I and P_O, the mean membership time is determined as

$$E_{m,m}(T_m) = \; P_I(\tau_{2,I} + \tau_{1,I} + E_{s,s}(T_m)) + P_O(\gamma\tau_{2,E} + (1-\gamma)(\tau_{2,I} + \tau_{1,E})) \tag{4.19}$$

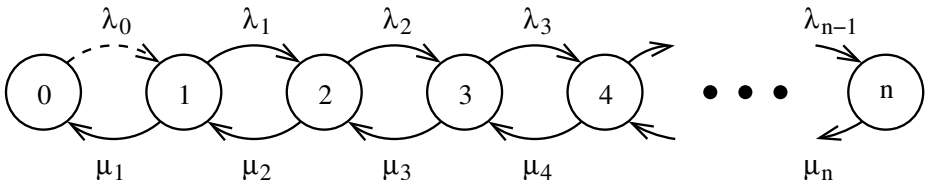


Figure 4.3: The Markov transition diagram.

where $E_{m,m}(T_m)$ denotes the mean of T_m in the case that both nodes are moving initially.

Summarizing all the three cases, the mean cluster membership time can be written as

$$E(T_m) = P_{s,s}E_{s,s}(T_m) + P_{s,m}E_{s,m}(T_m) + P_{m,m}E_{m,m}(T_m), \tag{4.20}$$

where $P_{s,s}$, $P_{s,m}$, $P_{m,m}$ are the probabilities of the respective cases. We know from the mobility model that the mean duration is τ_s in the pausing phase and $\tau_m = E(\frac{D}{V}) = E(D)E(\frac{1}{V}) = \frac{d_{max}}{2}E(\frac{1}{V})$ in the moving phase. Therefore, the probabilities of a node's mobility phases are obtained as

$$P_s = \frac{\tau_s}{\tau_s + \tau_m}, \qquad P_m = \frac{\tau_m}{\tau_s + \tau_m}, \tag{4.21}$$

from which we further have

$$P_{s,s} = P_s^2, \quad P_{s,m} = 2P_sP_m, \quad P_{m,m} = P_m^2. \tag{4.22}$$

We determine next the cluster lifetime T_h, which is the time duration before all the clustermembers leave the clusterhead. We use a Markov model to study its mean value. The Markov transition diagram is depicted in Fig. 4.3, in which a state is defined to be the number of clustermembers affiliated to the clusterhead. We assume the coming and leaving of clustermembers obey Poisson processes. The transitions between states are viewed as a birth-death process. Each time a node joins the cluster, the state transits one step to the right; each time a clustermember leaves the cluster, the state transits one step to the left. At time zero, the transition starts at an initial state S_{init}, which denotes the number of clustermembers at the time the cluster is constructed. The cluster lifetime is the transition time from state S_{init} to S_0.

On state S_0 the clusterhead will recluster and try to merge into another cluster, so the clustermember joining rate $\lambda_0 = 0$. The joining rate λ_j on the other states S_j $(1 \leq j \leq n-1)$ is state dependent with finite node population. For any two states S_{j_1} and

S_{j_2}, $\lambda_{j_1} > \lambda_{j_2}$ if $j_1 < j_2$. To simplify the model, we truncate the Markov chain at the state S_n, where $n \ll N$. In this truncated model $\lambda_{j_1} \approx \lambda_{j_2} = \lambda$. We observe from simulations that the cluster size is always within a range far less than the node population N, so we set n to be a few multiples larger than the initial number of clustermembers. We use 5 multiples in our computations. λ is determined as follows. Let N_m and N_h denote the total number of clustermembers and clusterheads in the network respectively. We have

$$\lambda_j \approx \lambda = \Big(\frac{N_m}{E(T_m)} + \frac{N_h}{E(T_h)}\Big) \cdot \frac{\pi r^2}{l^2} \cdot \beta_1 \cdot \beta_2 \quad (1 \le j \le n-1), \tag{4.23}$$

where $\frac{N_m}{E(T_m)} + \frac{N_h}{E(T_h)}$ accounts for the networkwide arrival rate of nodes seeking a cluster to join, $\frac{\pi r^2}{l^2}$ is the geographical factor considering the percentage that takes place in the clusterhead's transmission area, β_1 is the mobility factor, and β_2 is the clusterhead selection factor. β_1 is concerned with the fact that a node is able to join the clusterhead only when they have relative movement: a relative stationary node inside the clusterhead's neighborhood is likely to be an affiliated clustermember already and a relative stationary node out of the reach of the clusterhead does not have the chance to join this clusterhead. β_2 considers the chance for a node to join a particular clusterhead given the possibilities that it may see several candidate clusterheads in its neighborhood and it may also become a clusterhead itself. We determine β_1 and β_2 as

$$\beta_1 = P_m + P_s P_m, \qquad \beta_2 = \frac{1}{\frac{\pi r^2}{l^2} N_h + 1}. \tag{4.24}$$

The clustermember departure rate is also state dependent and expressed as

$$\mu_j = j\mu, \qquad \mu = \frac{1}{E(T_m)} \qquad (1 \le j \le n). \tag{4.25}$$

The initial state S_{init} has $\frac{N_m}{N_h}$ number of clustermembers on average.

The difficulty in finding $E(T_h)$ rises from the fact that there exists infinite number of possible paths from S_{init} to S_0. We solve this problem by using an equation array. Defining t_j $(0 \le j \le n)$ as the mean transition time from S_j to S_0, we have

$$\begin{cases} t_0 = 0, \\ t_j = \frac{1}{\lambda_j + \mu_j} + \frac{\mu_j}{\lambda_j + \mu_j} t_{j-1} + \frac{\lambda_j}{\lambda_j + \mu_j} t_{j+1} \quad (1 \le j \le n-1), \\ t_n = \frac{1}{\mu_n} + t_{n-1}. \end{cases} \tag{4.26}$$

The transition time from a state S_j to S_0 is the sum of the sojourn time on state S_j and the transition time from the successive state (S_{j-1} or S_{j+1}) to S_0. The sojourn time is

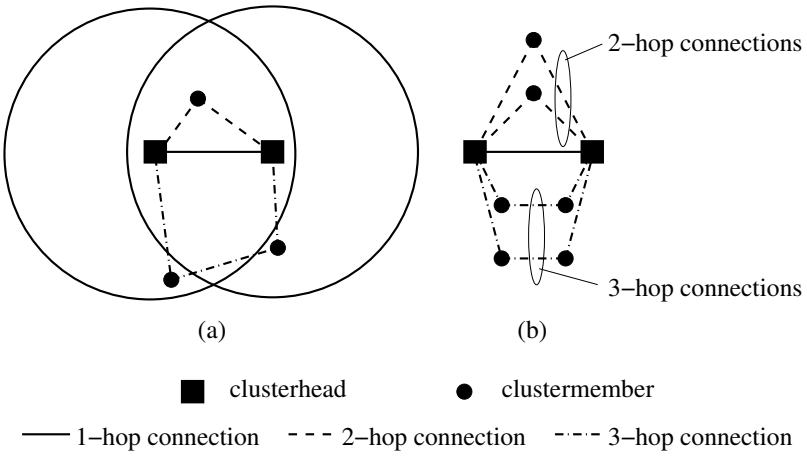


Figure 4.4: The connection types and the logical link.

determined by when a joining node arrives or an affiliated clustermember leaves. Since both are Poisson processes, the combination is a Poisson process with parameter $\lambda_j + \mu_j$. Therefore the mean sojourn time on S_j is $\frac{1}{\lambda_j+\mu_j}$. After this sojourn time, a left transition occurs with probability $\frac{\mu_j}{\lambda_j+\mu_j}$ or a right transition occurs with probability $\frac{\lambda_j}{\lambda_j+\mu_j}$. At state S_n, only the left transition is possible since there are no arrivals of joining nodes. Solving (4.26) for all the t_j, we have $E(T_h) = t_{init}$. Note that (4.23) requires the knowledge of $E(T_h)$. In the computations, $E(T_h)$ is first assigned an initial value and then computed recursively until it converges.

4.3.2 Inter-Cluster Link Lifetime

In order to establish a routing backbone with the clusterheads, neighbor clusterheads must be connected. The inter-cluster connectivity is determined by the availability of the links that connect two nodes from the two clusters respectively. Such links include those between the two clusterheads, one clusterhead and one clustermember, and two clustermembers. They provide direct or indirect connectivity between the two clusterheads. Fig. 4.4(a) illustrates all the connection types. We define the inter-cluster *Logical Link* to be the set of all these connections, as shown in Fig. 4.4(b). The logical link exists as long as *any* of the connection types exists. The *longest* inter-cluster link lifetime is then the lifetime of the logical link.

When a logical link consists of multiple connection types, out of the efficiency

consideration, a routing protocol preferably chooses the shortest one to transmit packets. Therefore, if a clustermember can be used in both a 2-hop connection and a 3-hop connection, the 2-hop connection will always be used. This observation allows us to assume that a clustermember does not appear in more than one connections, which further leads to the assumption that all the inter-node links used in a logical link are different and independent. The mean cluster membership time $E(T_m)$ derived earlier describes the neighboring duration between a clusterhead and a clustermember. It can be generalized as the mean neighboring duration between any two neighbor nodes. Thus, following our assumption in Section 4.3.1, the inter-node link lifetimes have identical and exponential distributions with mean $\frac{1}{\mu} = E(T_m)$.

The lifetime of the logical link is determined by the link's initial composition, the arrivals of new connections and the failures of existing connections. Due to the unmanageable difficulty in determining the new connection arrival process, we approximate the logical link lifetime with the time to the failures of all the initial connections, which serves as a lower estimate of the real lifetime. Let n_i $(i = 1, 2, 3)$ denote the initial number of i-hop connections, $\mathcal{F} = \{F_x^{(j)}\}$ $(j = 1, 2, \cdots, \sum_{i=1}^{3} n_i)$ denote a permutation of the failure sequence of the $\sum_{i=1}^{3} n_i$ connections where $F_x^{(j)} \in \{F_1, F_2, F_3\}$ denotes that the j-th failure happens on an x-hop connection, $n_i^{(j)}$ denote the number of remaining i-hop connections after the $(j-1)$-th but before the j-th failure in the sequence $\mathcal{F}$, and $1_i(F_x^{(j)})$ denote the indicator functions such that

$$1_i(F_x^{(j)}) = \begin{cases} 1 & x = i \\ 0 & x \neq i \end{cases} \qquad (i = 1, 2, 3). \tag{4.27}$$

The mean lifetime of the logical link is written as

$$E(T_l) = \sum_{\{\mathcal{F}\}} P(\mathcal{F}) T(\mathcal{F}), \tag{4.28}$$

where $P(\mathcal{F})$ is the probability of $\mathcal{F}$, $T(\mathcal{F})$ is the mean lifetime given $\mathcal{F}$, and $\{\mathcal{F}\}$ has cardinality $\binom{n_1+n_2+n_3}{n_1} \cdot \binom{n_2+n_3}{n_2} \cdot \binom{n_3}{n_3} = \frac{(n_1+n_2+n_3)!}{n_1! n_2! n_3!}$. $P(\mathcal{F})$ and $T(\mathcal{F})$ are determined by

$$P(\mathcal{F}) = \prod_{j=1}^{n_1+n_2+n_3} P(F_x^{(j)}) = \prod_{j=1}^{n_1+n_2+n_3} \frac{\sum_{i=1}^{3} i \cdot n_i^{(j)} \cdot 1_i(F_x^{(j)})}{\sum_{i=1}^{3} i \cdot n_i^{(j)}}, \tag{4.29}$$

$$T(\mathcal{F}) = \sum_{j=1}^{n_1+n_2+n_3} T(F_x^{(j)}) = \sum_{j=1}^{n_1+n_2+n_3} \frac{1}{\sum_{i=1}^{3} i \cdot n_i^{(j)} \cdot \mu}. \tag{4.30}$$

Table 4.1: Mobility Patterns

Pattern	τ_s (min)	$[v_{min}, v_{max}]$ (m/s)	d_{max} (m)
MP1	8	$[1, 3]$	1000
MP2	6	$[1, 5]$	1000
MP3	4	$[1, 7]$	1000
MP4	2	$[1, 9]$	1000

In (4.29), $\sum_{i=1}^{3} i{\cdot}n_i^{(j)}$ is the total number of inter-node links before the j-th connection failure, $\sum_{i=1}^{3} i \cdot n_i^{(j)} \cdot 1_i(F_x^{(j)})$ is the number of inter-node links of which any break will result in $F_x^{(j)}$, their ratio determines $P(F_x^{(j)})$, and $P(\mathcal{F})$ is the multiplication of all the $P(F_x^{(j)})$'s. In (4.30), $\frac{1}{\sum_{i=1}^{3} i{\cdot}n_i^{(j)}{\cdot}\mu}$ is the mean time between the $(j-1)$-th and the j-th connection failures, and $T(\mathcal{F})$ is the sum of all these intervals.

4.3.3 Path Lifetime

A path in the hierarchical network architecture is composed of a sequence of inter-cluster links. Considering the clustering roles of the two end nodes in a communication session, there are also possible intra-cluster links at the two ends of the path. If the node at one end of the path is a clustermember, there must be a clustermember–clusterhead link at that end of the path. The lifetime of the path is determined by the shortest lifetime among all these links. Assuming that the inter-cluster and the intra-cluster links all have exponentially distributed lifetimes, with the *longest* mean $E(T_l)$ and $E(T_m)$ respectively, the *longest* path lifetime is then also exponentially distributed with the mean determined by

$$
\begin{aligned}
E(T_p) &= P_{h\text{-}h}E_{h\text{-}h}(T_p) + P_{h\text{-}m}E_{h\text{-}m}(T_p) + P_{m\text{-}m}E_{m\text{-}m}(T_p), \\
P_{h\text{-}h} &= (\frac{N_h}{N})^2, \quad P_{h\text{-}m} = \frac{2N_hN_m}{N^2}, \quad P_{m\text{-}m} = (\frac{N_m}{N})^2, \\
E_{h\text{-}h}(T_p) &= \frac{1}{k\mu_l}, \quad E_{h\text{-}m}(T_p) = \frac{1}{k\mu_l + \mu}, \quad E_{m\text{-}m}(T_p) = \frac{1}{k\mu_l + 2\mu},
\end{aligned} \tag{4.31}
$$

where $P_{h\text{-}h}$, $P_{h\text{-}m}$, $P_{m\text{-}m}$ are the probabilities of the end node roles, $E_{h\text{-}h}(T_p)$, $E_{h\text{-}m}(T_p)$, $E_{m\text{-}m}(T_p)$ are the mean path lifetimes in the respective cases, k is the number of inter-cluster links in the path, $\mu_l = \frac{1}{E(T_l)}$, and $\mu = \frac{1}{E(T_m)}$.

4.3.4 Numerical Results

Due to problem complexity, accurate analytical determination of the parameters N_h, N_m, n_i $(i = 1, 2, 3)$ and c is difficult. We use simulations to obtain their average values and then apply them in the analysis. We configure an example network as $N = 240$, $l = 2000$ meters, $r = 250$ meters, and specify four mobility patterns by tuning the node moving speed and pause time as shown in Table 4.1 to represent different mobility scenarios, such as low speed pedestrians and high speed vehicles. Besides the theoretical analysis, we have also implemented the Lowest-ID and the GDMAC ($K = 3$, $H = 32$ as in [74]) clustering algorithms as comparison in the NS-2 simulator [98].

The mean cluster membership times from the analysis and the simulations are shown in Fig. 4.5. We observe that the Lowest-ID has a quite short cluster membership time as compared to the theoretical bound. The GDMAC improves over the Lowest-ID, but there is still a noticeable gap from the upper bound obtained from analysis. Similar observation on the mean cluster lifetime is shown in Fig. 4.6. We determine N_h and N_m by letting each node stay in its role (clusterhead/clustermember) as long as its cluster is still valid. That is, a clustermember tries to re-cluster only after it has lost the contact to its clusterhead and a clusterhead tries to re-cluster only after it has lost the contact to all of its clustermembers. The average values measured are $N_h = 56$ and $N_m = 184$ across all the four mobility patterns with very slight variations (± 2). The transition time t_j is then determined by (4.26) and $t_{init} = t_3$ because S_3 is the state closest to the mean cluster size $\frac{184}{56} = 3.28$. We see in Fig. 4.6 that the cluster lifetime of the Lowest-ID is significantly lower than the analytical bound and the GDMAC does not reach this bound yet.

With $N_h = 56$ and $N_m = 184$, we measure the average number of connections between neighbor clusterheads and obtain $n_1 = 0.23$ (± 0.03), $n_2 = 1.85$ (± 0.1), $n_3 = 1.05$ (± 0.15) across the four mobility patterns. As they are not integers, we approximate them as:

$$n_1 = \begin{cases} 1 \ w.p.\ 0.23 \\ 0 \ w.p.\ 0.77 \end{cases}, \qquad n_2 = \begin{cases} 2 \ w.p.\ 0.85 \\ 1 \ w.p.\ 0.15 \end{cases}, \qquad n_3 = \begin{cases} 2 \ w.p.\ 0.05 \\ 1 \ w.p.\ 0.95 \end{cases}.$$

The mean inter-cluster link lifetime is then determined from (4.28). Fig. 4.7 plots the results from analysis and simulations. Again we observe the similar instability of the Lowest-ID in the inter-cluster connectivity and the stability gap between the GDMAC and the analytical bound.

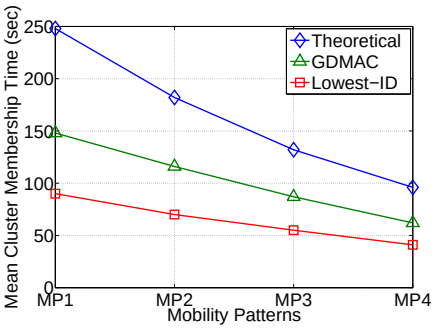


Figure 4.5: $E(T_m)$.

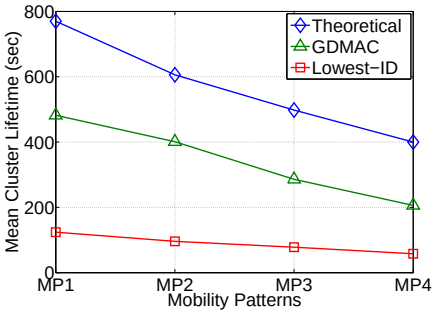


Figure 4.6: $E(T_h)$.

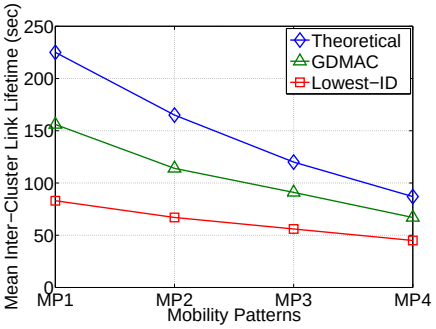


Figure 4.7: $E(T_l)$.

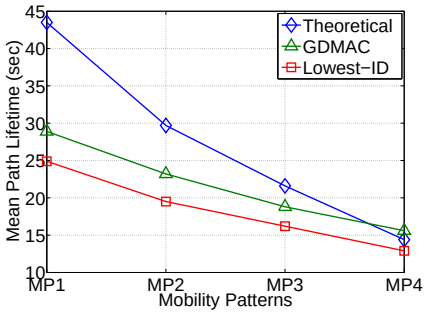


Figure 4.8: $E(T_p)$.

Finally we choose 10 random source-destination pairs and measure their average path length in the number of inter-cluster links with $N_h = 56$ and $N_m = 184$. The measured average path lengths are $c = 3.95, 4.21, 4.21, 4.71$ respectively in the four mobility patterns. The same 10 node pairs are also used in the Lowest-ID and the GDMAC simulations. The analytical mean path lifetime is determined by (4.31) and plotted in Fig. 4.8 together with the simulation results. Fig. 4.8 presents that the path lifetime could be higher than those achieved by the Lowest-ID and the GDMAC. We notice that in the mobility pattern 4 the path lifetimes from the analysis, GDMAC, and Lowest-ID are very close, and GDMAC even exceeds the analytical result. This is due to the fact that our analysis derives a lower estimate of the inter-cluster link lifetime, as mentioned in Section 4.3.2, which leads to the underestimation of the path lifetime. However, this observation indeed indicates that there is limited room for improving the path stability in high mobility environment.

4.4 The Mobility and Energy Aware Clustering Algorithm

The analysis reveals that the maximum topological stability is achieved by extending a node's clustering status to its natural ending. Premature reclustering terminates the valid clusters and therefore undermines the network topological stability. In this section we design a new clustering algorithm that aims to achieve the longest cluster lifetime. This new algorithm has two features: first, it preserves the existing clusters to the maximum extent; second, it also tries to maximize the cluster lifetime at the time of cluster construction. The first feature is straightforward in the design. Once a cluster is constructed, node joining and leaving are allowed, but as long as the cluster is still valid a coming node cannot force the clusterhead to revoke its role or a clustermember to switch to a different cluster. The second feature is implemented as choosing the most stable nodes to become the clusterheads. A node is stable if it stays in the clusterhead status for long time. We interpret the stableness mainly in the mobility sense but also with the energy consideration. As such, we name the new algorithm the Mobility and Energy Aware Clustering Algorithm (MEACA).

We define two node attributes for cluster construction. The mobility attribute of a node u is defined to be $A_m = \sum_{v \in N(u)} \tau_v$, where $N(u)$ is the neighbor set of u, τ_v is the neighboring time between a neighbor node v and the node u. The neighboring time is acquired through the use of *hello* messages. Each node periodically broadcasts hello messages to inform the neighbors of its presence. Node u determines the duration that it has been in contact with v by summing up all the continuous time ticks on which it has heard from v. The attribute A_m indicates a node's relative mobility to its neighbors: larger value means higher stableness. The energy attribute A_e of u is defined to be its estimated lasting time of the remaining energy. Each node exchanges its attributes with every neighbor node through the hello messages.

When a node u first joins the network or needs reclustering, it determines its clustering status by comparing its own attributes with its neighbors' attributes. It finds the node that has high A_m and A_e values to be its clusterhead. If this node is itself, it becomes a clusterhead; otherwise, it becomes a clustermember to join the selected node. Fig. 4.9 presents the clusterhead selection algorithm in pseudocode. The node u checks periodically if it needs reclustering. If its status is a clusterhead but there is no affiliated clustermember any more or its status is a clustermember but it has lost contact to its clusterhead, the clustering algorithm is executed. In the first step, u adds itself to the list of nodes from

The MEACA Algorithm

```
1   initialize u.timer
2   when u.timer expires
3     reschedule u.timer
4     if   u.status = HEAD and u.member ≠ NULL
        or u.status = MEMBER and u.head ≠ NULL
5       then return
6     C ← N(u) ∪ {u}
7     C ← {v | v.status ≠ MEMBER, v ∈ C}
8     A*_m ← α · max{A_m(v) | v ∈ C} (0.9 < α < 1)
9     S ← {v | A_m(v) ≥ A*_m, v ∈ C}
10    w ← arg_v max{A_e(v) | v ∈ S}
11    if w = u
12      then u.status ← HEAD
13           u.head ← u
14    else if w.status = HEAD
15      then u.status ← MEMBER
16           u.head ← w
17    return
```

Figure 4.9: The MEACA clustering algorithm.

which the clusterhead will be selected. Then it excludes any node that has become a clustermember, because it cannot take the clusterhead status according to our requirement of preserving the existing clusters. Next, it finds the maximum mobility attribute of the nodes in the list and determines a mobility stableness threshold A_m^*, where α is a constant parameter. The purpose of α is to prevent the selection of a clusterhead that has low energy even though it may have the highest mobility stableness. We limit α to the range $(0.9, 1)$ for the algorithm convergence considerations. After A_m^* is determined, it shortlists the nodes whose mobility attributes exceed A_m^*. Finally, it finds the node w that has the highest A_e in the shortlist to be its clusterhead. If $w = u$, u becomes a clusterhead; if $w \neq u$, u registers its membership with w. Since w may also be in the clustering process, there are three possibilities. If w has determined to be a clusterhead, w accepts u's registration. If w has determined to be a clustermember, w rejects u's registration and u repeats the clustering algorithm. In this second round as w is excluded from the set C, u will end up with another selection of the clusterhead. If w has not made its decision yet, u waits until w has made the decision.

After u has determined its clustering status, it stays in this status until either of

the two following situations takes place: 1) if u is a clusterhead, all of its clustermembers have left the cluster; 2) if u is a clustermember, it has left its selected cluster. In these cases, u determines its new status again.

MEACA has the following important properties: 1) the algorithm converges, 2) the algorithm has no ripple effect, and 3) the algorithm balances node energy.

The algorithm convergence is the property that a node is able to determine its status in finite time. We prove this property in different cases. *Case 1:* if u determines to be a clusterhead, its status is finalized in one round of executing this algorithm. *Case 2:* if u selects w ($w \neq u$) and w is a clusterhead, u's status as a clustermember is finalized in one round too. *Case 3:* if w is a clustermember, u repeats the clustering process one more time with w excluded from its candidate list. In the worst case u excludes every node in $N(u)$ and ends up with itself. This takes $|N(u) \cup \{u\}|$ rounds. *Case 4:* if w has not made its decision yet but it is not waiting for another node to make decision, then w will determine its status in finite number of rounds according to our discussion on u and afterwards u will finalize in finite number of rounds too. *Case 5:* if w is waiting for w', the chain of waiting nodes $u, w, w', \cdots$ must be finite given the finite node population in the network and they satisfy the inequality $A_m(u) < A_m(w) < A_m(w') < \cdots$ with high probability when α is large. The last node in this chain will determine its status first and afterwards the rest nodes will determine their status successively in the reverse order of the chain with finite number of clustering rounds each. *Case 6:* if $A_m(u) > A_m(w) > A_m(w') > \cdots$ happens (due to $\alpha < 1$) and the waiting nodes form a loop, node movements will break the loop.

The ripple effect is the phenomenon that a re-clustering node triggers a sequence of cluster changes in a wide area. In MEACA, a re-clustering node may join an existing clusterhead or construct a new cluster, but it does not force the existing clusterhead to revoke its clusterhead status or an existing clustermember to switch to it. Thus re-clustering is limited to the participating nodes only and the rest of the network is not affected.

From a fairness point of view, it is ideal to rotate the clusterhead roles among the nodes because the clusterheads consume energy faster than the clustermembers. In MEACA, the node with the highest energy becomes the clusterhead when several candidate nodes have comparable mobility stableness. A former clusterhead is expected to have lower energy than the former clustermembers and therefore likely to hand over the clusterhead role to others at the time of re-clustering. As the mean cluster lifetime is relatively short (minutes) as compared to the usual node energy lasting time (hours), a clusterhead has

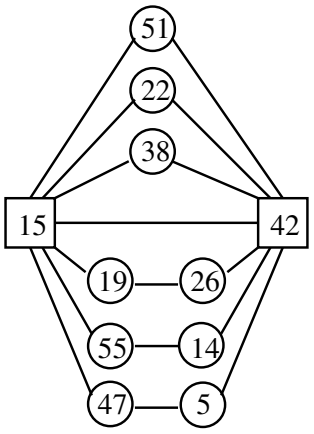
plenty of chances to shift its role before depleting its energy.

The main differences of MEACA from the existing work on clustering algorithms are in two aspects. First, the existing algorithms [68, 73, 74] are *extending* rather than *maximizing* the cluster stability, while MEACA attempts to reach the maximum cluster stability under the constraint of node mobilities. As the analysis suggests, the maximum cluster lifetime is achieved in mobile networks when all the unnecessary cluster changes are avoided. MEACA maximizes the cluster stability by eliminating the premature re-clusterings. Second, when node mobility is considered, the existing schemes [67, 70, 71] measure the node speed first and then infer the network topology stableness, while MEACA directly measures the topology stableness by defining the mobility attribute as the node neighboring times, which is simpler but more accurate.

4.5 The Clustered Network Overlay Routing Protocol

From the analysis we know that in order to achieve the longest path lifetime it is equally important to maximize the inter-cluster connectivity besides stabilizing individual clusters. In this section we propose the Clustered Network Overlay Routing Protocol (CNORP) which aims to achieve two goals: 1) establishing the inter-cluster links and maintaining them to the maximum extent; 2) implementing the path discovery and packet forwarding mechanisms in the overlay network composed of the connected clusterheads.

We have shown in Fig. 4.4 the three types of connections between two neighbor clusterheads and defined the inter-cluster logical link. The CNORP protocol implements the logical link and keeps it up-to-date. It avoids unnecessary path rediscovery when a connection in the logical link breaks as long as alternative connections are available. A clusterhead gathers the connection information through *hello* messages. These hello messages are different from those used in the clustering attribute exchanges, though they may be combined. Every node periodically broadcasts hello messages which carry the ID of its clusterhead and the distance from its clusterhead. If the node is a clusterhead, the distance is 0; if it is a clustermember, the distance is 1. A clusterhead that receives the broadcast from its neighbors then knows the 1-hop and 2-hop connections to its neighbor clusterheads: distance 0 in a received hello message indicates a 1-hop connection to the broadcasting clusterhead; distance 1 indicates a 2-hop connection via the broadcasting clustermember. In addition to the hello messages, a clustermember also periodically reports its received hello

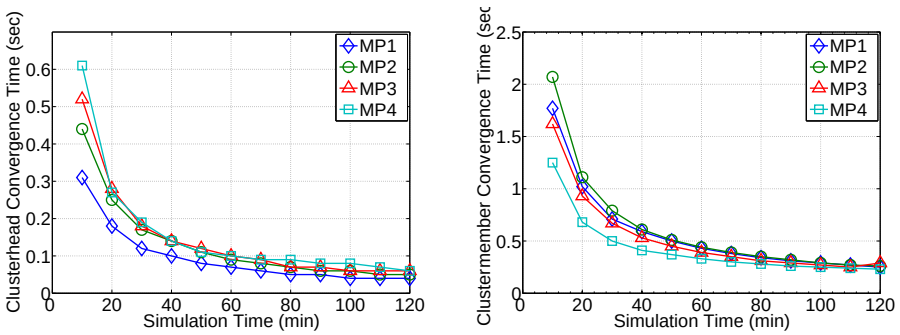


Node 15's Connection Table		
ID	Type	Next Hop
42	1−hop	42
	2−hop	22 38 51
	3−hop	19 47 55

Figure 4.10: An example connection table.

messages to its clusterhead. These reports provide supplementary connection information: if the clustermember has a 1-hop or a 2-hop connection to a neighbor clusterhead, then its clusterhead may utilize it to reach that neighbor clusterhead in a 2-hop or a 3-hop connection via the reporting clustermember. A clusterhead organizes the connections in a table format as shown in Fig. 4.10. The connection table is kept up-to-date as any topology changes will be reflected in the periodical hello messages and clustermember reports. An inter-cluster logical link is broken at the time that all the three connection type entries become empty. When a clusterhead forwards packets to a neighbor clusterhead, it always chooses the shortest available connection in the table. If there are multiple shortest connections, a random selection is made among them. A clustermember organizes its connections to the neighbor clusterheads in a similar table. Because a clustermember does not receive reports from other clustermembers, it does not have 3-hop connections. The table is kept up-to-date by the periodically received hello messages. When a clustermember forwards packets to a neighbor clusterhead, it uses the shortest connection in the table.

In the overlay network of the connected clusterheads any reactive routing protocol, such as AODV and DSR, can be used for the path discovery and maintenance. Proactive routing protocols are not suitable because of their excessive overhead when the clusters change. During path discovery, route request broadcasting is achieved in a unicast way. If a clusterhead needs to broadcast, it unicasts a copy of the message to each neighbor clusterhead via the shortest connection in the respective logical link. A path discovered is a sequence of clusterheads from the source node to the destination node. If the source



(a) Clusterhead (b) Clustermember

Figure 4.11: The node status determination time (MEACA).

and the destination nodes are clustermembers, the source node's packets are sent to its clusterhead first, then routed to the destination node's clusterhead, and finally delivered to the destination node. An intermediate clusterhead is flexible in choosing a connection to forward a packet to the next clusterhead. The selection is determined by the availability of the connections at the time of packet forwarding, but the use of the shortest available connection is always enforced.

The CNORP protocol differs from the existing work in that it does not require a global cluster membership table as compared to [73, 76] and it simplifies path management and maximizes path lifetime as compared to [77, 78] by utilizing the inter-cluster logical links. In comparison to [79], a clustermember using the CNORP protocol acquires sufficient local routing information by itself such that it is able to forward packets independently to the neighbor clusterheads without obtaining the routing information from its clusterhead. In addition, the use of unicast to implement route request broadcasting suppresses the redundant broadcast messages in a simpler way.

4.6 Performance Evaluation

We have implemented MEACA and CNORP in NS-2 for performance evaluation. The same simulation settings as in Section 4.3.4 are used. We set $\alpha = 0.9$ and the frequency of neighbor node information exchange to be every 2 seconds.

Fig. 4.11(a) plots the mean time spent by a node in determining its status as a

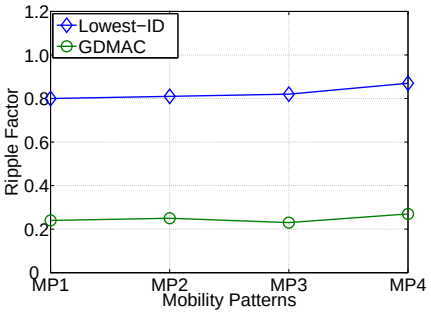
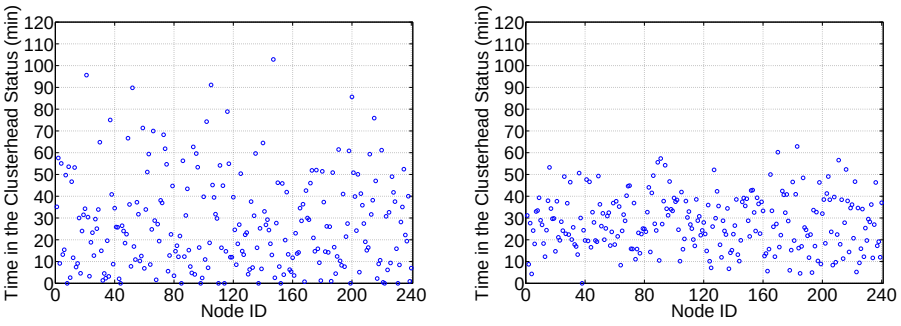


Figure 4.12: The ripple effect of Lowest-ID and GDMAC.

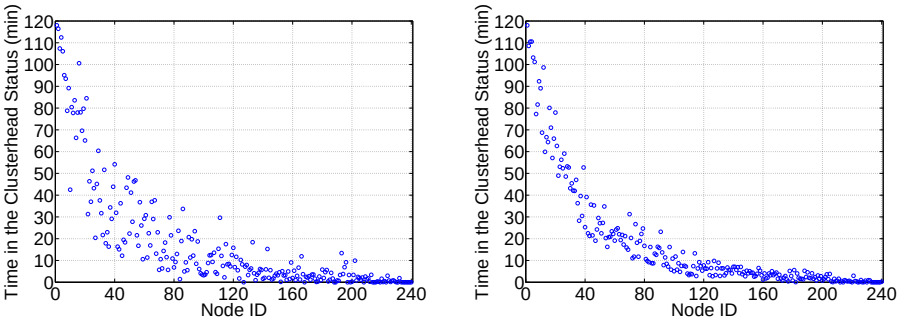


(a) MP1 (b) MP4

Figure 4.13: The time distribution of the clusterhead status (MEACA).

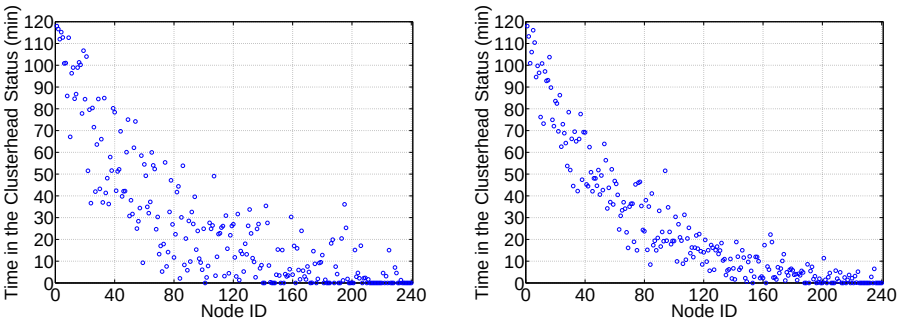
clusterhead using the MEACA algorithm. Initially it takes 0.3–0.6 second for a node to become a clusterhead, because at the beginning of the simulation all the nodes are trying to determine their roles and their decisions are interdependent. Some nodes must wait to determine until their neighbors have finalized. As the simulation proceeds, the mean time decreases to be less than 0.1 second. The similar trend is observed in the mean time to determine the clustermember role, which decreases from 1.3–2 seconds to 0.3 second as shown in Fig. 4.11(b). These results demonstrate that the MEACA algorithm converges.

We have proven that the MEACA algorithm does not have the ripple effect. As comparison, we plot the ripple effect of the Lowest-ID and the GDMAC in Fig. 4.12, where the *ripple factor* is defined to be the ratio of the number of premature re-clustering to the number of matural re-clustering. The Lowest-ID has a ripple factor of 0.8, indicating



(a) MP1 (b) MP4

Figure 4.14: The time distribution of the clusterhead status (Lowest-ID).



(a) MP1 (b) MP4

Figure 4.15: The time distribution of the clusterhead status (GDMAC).

that each node status update causes on average extra 0.8 update among its neighbors. The GDMAC still has the ripple effect problem though it reduces the ripple factor to 0.2.

Fig. 4.13, 4.14, 4.15 depict the cumulative time of staying in the clusterhead status for each node during a 2-hour simulation. We observe obvious dependence of this cumulative time on node ID numbers in the Lowest-ID and the GDMAC algorithms, where the nodes with low IDs take the clusterhead status significantly longer than the others. This phenomenon poses big fairness and energy balancing problems. The MEACA algorithm, on the contrary, does not have the ID-dependence problem. We see in Fig. 4.13 that both the low-ID and the high-ID nodes have the equal chance to become the clusterheads.

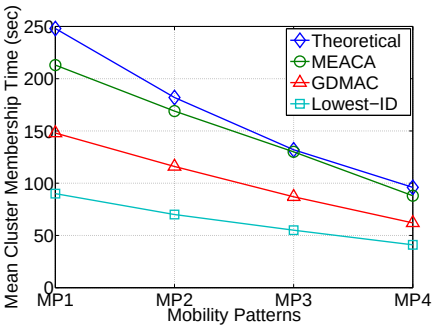


Figure 4.16: $E(T_m)$.

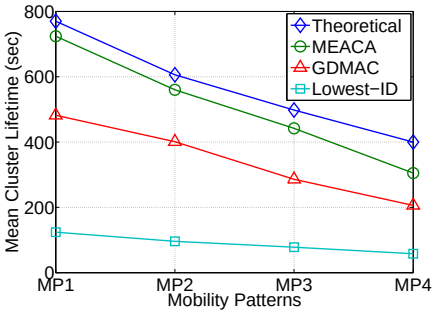


Figure 4.17: $E(T_h)$.

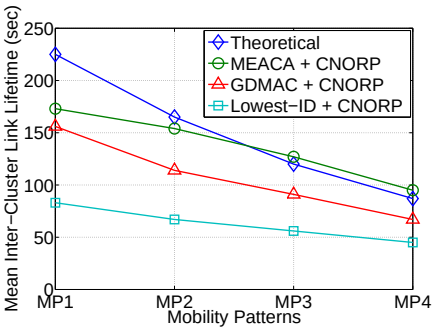


Figure 4.18: $E(T_l)$.

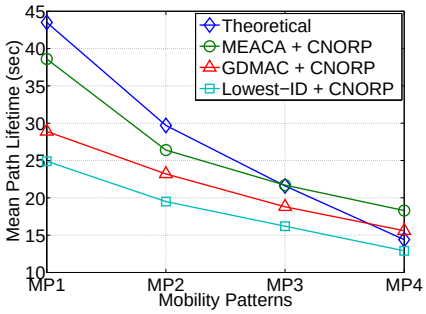


Figure 4.19: $E(T_p)$.

Fig. 4.16 and 4.17 compare the cluster membership time and the cluster lifetime achieved by MEACA to the previous results obtained from analysis and the Lowest-ID and GDMAC simulations. We observe that MEACA improves over the Lowest-ID and the GDMAC, and its performance approaches the theoretical bounds. Similar improvements are observed in Fig. 4.18 and 4.19 that compare the inter-cluster link lifetime and the path lifetime. We see that the performance of MEACA and CNORP exceeds the theoretical results with high mobility (MP3 and MP4), which is due to the underestimation of inter-cluster link lifetime and path lifetime in the analysis, as mentioned earlier. Finally, Fig. 4.20 shows the normalized throughout of the 10 random connections using different clustering algorithms. As a result from longer path lifetime, MEACA achieves higher end-to-end throughput than the Lowest-ID and the GDMAC algorithms.

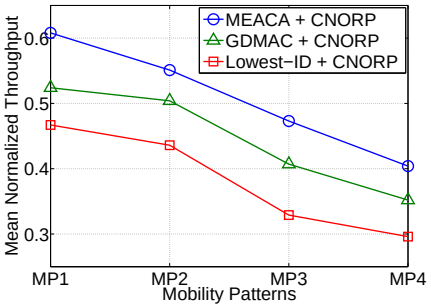


Figure 4.20: Throughput comparison.

4.7 Summary

In this chapter we have presented the mathematical modeling and analysis of the maximum lifetimes of clusters, inter-cluster links and end-to-end paths in a mobile environment described by a Random-Walk-like mobility model. We have found that the maximum topological stability of hierarchical architecture is achieved when premature re-clustering is avoided and inter-cluster connectivity is maximized. Enlightened by the analysis, we have designed the MEACA clustering algorithm and the CNORP hierarchical routing protocol that work together to maximize the topological stability of hierarchical networks. Simulation shows that their performance approaches the theoretical bounds obtained from analysis.

Chapter 5

The Wireless Network Resilience against Correlated Failures

Besides node mobility, node failure is another commonly observed threat to large-scale wireless networks. Especially when the failures are correlated, the impact of node failures is potentially devastating, because an initial local failure may trigger a global sequence of related failures. We characterize the spread of correlated failures in this chapter, which lays the foundation for evaluating and improving the failure resilience of existing wireless networks. We model the failure contagiousness as two generic functions: the failure impact radius distribution function $f(x)$ and the failure connection probability function $g(x)$. By using the percolation theory, we determine the respective characteristic regimes of $f(x)$ and $g(x)$ in which correlated failures will and will not percolate in the network. As our model represents various failure scenarios, the results are generally applicable in understanding the spread of a wide range of correlated failures.

5.1 Motivation and Related Work

Failures are always a potential risk in complex computer and communication networks. In a complex network a node may fail in many circumstances, for example, design mistakes, unexpected operations, malicious alterations, and power outages. In particular, failures happen inevitably in large wireless networks due to a number of reasons that include the limited battery supply for each node, physical exposure of the devices to potential adversaries, and dynamic environment that may interrupt communications and isolate nodes

from one another. When failures occur, the network suffers from degraded performance because of the unavailability of the failed nodes. The subsequent impact could range from insignificant topological changes to devastating network shutdown, depending on the role criticality of the failed nodes in the system.

Given the fact of failure inevitability and its detrimental consequences, many studies have been dedicated to the design of effective solutions that prevent [99, 100], detect [101, 102], mitigate [103, 104] and repair [105, 106] failures to retain normal network functioning. The majority of these studies deal with specific types of failures and their countermeasures, for example, planning traffic paths away from the failure-prone regions [100], providing continuous surveillance when a subset of sensors fail [105], finding alternative routes when nodes or links become unavailable [106, 107], and using small-sized packets [108] and error control codes [109] to communicate in networks with low-quality links. These studies have greatly contributed to the failure resilience of large networks.

Different from the existing studies, we are interested in quantifying the topological impact of failures. Instead of inspecting the details of a particular failure, we intend to determine the extent of composition and structure changes in the network after failure occurrence. We note that the impact created by a node failure depends on the functionality and sometimes the location of the node. It is evident that a network suffers the greatest difficulty in fulfillment of its communication tasks if the core functional or pivotally located nodes stop working properly. However, it is also observed in practice that large networks are usually designed with the single-point-of-failure problem considered and all the nodes undertake similar responsibility to avoid the worst-case failure scenario. We hence assume in our research that every node is identical and the failure impact on the network is soly determined by the amount of failed nodes.

In the literature, the impact of failures has been studied with the assumption that failures are random and independent. In [110], Xing and Wang analyzed the critical phase transition time of a large wireless network in which random failures gradually break down an initially connected network into small pieces of disjoint components. Their results present an insightful observation on the impact of failures from the network connectivity perspective. However, the work in [110] does not characterize all the failure possibilities. In many network scenarios, causal relations exist among failures, i.e., some failures happen as a result of other earlier failures. One example of correlated failures is the node energy depletion resulted from traffic overloading [111]. When a node fails, the traffic that originally

passed through it must be redistributed to the neighboring nodes. Some neighbors may work under heavy traffic loads, resulting in early energy depletion and node failure. Another commonly observed case of failure correlation is virus propagation. Computer viruses not only sabotage the infected nodes, but also reproduce themselves to propagate the infection. In comparison to the random and independent node failures, correlated failures pose an even greater challenge for keeping a large wireless network in operational state.

In view of the potentially devastating threat from correlated failures, it is imperative for us to obtain an insightful understanding of the correlated failure spread in large wireless networks. We characterize the scope of failure spread to evaluate the resilience of large wireless networks against correlated failures and furthermore to guide our future efforts toward containing failures. In an effort to gain a generally applicable result, we model the failure correlations as general mathematical functions and determine their characteristic regimes in terms of the ability of an initial failure to permeate the network. We use the correlation functions to model the geometric constraints in failure propagation. For example, failures may not be able to propagate in one hop beyond certain radius and they may infect other nodes only probabilistically. Based on the correlation functions, we use the percolation theory [112, 113] to tackle the failure spreading problem.

Percolation theory provides a mighty tool for analyzing a wide range of contact-and-relay problems observed in the real world. It examines the connection details in a very small fraction of the network but infers the occurrence or non-occurrence of a global phenomenon regarding the passage of some event of our interest. Percolation theory is therefore particularly suitable for studying the failure spreading problem. By utilizing the concept of percolation, we determine analytically the pervasiveness of failure spread conditioned on the failure correlation functions. Intuitively, stronger correlations drive an initial failure to spread into a larger area than weaker correlations. Our results provide a quantified answer on the relation between failure correlations and failure pervasiveness in large wireless networks. The results hence help network designers and operators evaluate and improve the resilience of large wireless networks to correlated node failures.

5.2 Problem Formulation

In large networks, nodes fail due to a variety of reasons. As we intend to understand the general behavior of failure spreading when the failures are interdependent, we ignore

the specifics of particular kinds of failures and concentrate on an abstract and generic mathematical representation of the correlations among failures. In reality, two categories of failures are observed: one-time failures and persistent failures. We discuss briefly the difference between these two categories.

If we disregard the recovery of nodes after failure, some types of failures have only one chance to fail a given node, which we will refer as *one-time failures*. An example is the computer virus infection. Computer virus exploits the shared security vulnerability of a commonly used software to infect the nodes that have installed this software. If a node has installed this software, it fails at the first contact from a virus-carrying node. If not, it is immune to the first and all the onward attacks.

In other failure scenarios, we observe that a node is subject to multiple times of failures of the same kind, which we will refer as *persistent failures*. A failure is persistent if a node may fail each time one of its neighbors fails. Different from one-time failures, persistent failures happen not due to node vulnerability, but because of the inter-connection between nodes. One example is traffic overloading and energy depletion. When a node fails, the traversing traffic will be redistributed to the neighbors via dynamic routing. Some neighbors may fail subsequently after a short time when their energy is depleted by the excessive traffic loads. Obviously, a node has the risk of failure whenever it receives extra load from neighbors.

We will present a generalized mathematical formulation of these two failure categories and investigate the scopes of failure spread under various conditions. The study helps us gain an insightful understanding of the failure distribution in large wireless networks in correspondence to the failure dependence among nodes, which can be utilized to improve the network resilience against correlated failures.

5.2.1 Network and Failure Models

We consider a large wireless network consisting of n nodes in a region $\mathcal{A} = [-\frac{L}{2}, \frac{L}{2}]^2$ ($L \to \infty$). The number of nodes n is assumed to be a Poisson distributed random variable with constant density λ everywhere in the network. Let X_i ($1 \leq i \leq n$) denote the random location of node v_i that is uniformly distributed in the network, independent of n and any X_j ($i \neq j$). By definition [114], $\mathcal{H}_\lambda = \{X_1, \cdots, X_n\}$ is a homogeneous Poisson point process.

Node failures are interdependent. When a node fails due to some reason, the failure

may propagate to other nodes. We use the word "propagate" only to annotate the temporal and spatial relations of the failures occurring at different locations. There are not necessarily messages exchanged between failed nodes. There are, however, causal connections between the failures. A failure occurs subsequently after an earlier failure because the earlier failure must have resulted in certain change in the second node. The change is easily identified in some cases. For example, the previous node was infected by a virus and the virus was transmitted to the second node through the wireless channel between them. In other cases, the change is implicit, such as an increase of the traffic load at the second node after the first node has failed. Specifically, we define two types of failure correlations in this study, as described below.

In the first type, which we define as *one-time failures*, a node is subject to the failure only at the first contact from a failed node. The real-world example is the spread of computer virus. The result of an attempted virus infection is determined by whether the attacked node is vulnerable to the virus or not. If the attacked node is vulnerable, it fails at the first time it receives the virus. If not, it does not fail and will resist all the onward attacks.

In the second type, which we define as *persistent failures*, a node may fail subsequently every time an earlier failure occurs within its neighborhood. The real-world example is the traffic overloading failures. For example, a sensor network is deployed for continuous surveillance for a certain area and the battery in each sensor node is recharged periodically to ensure continuous functioning. In the normal case, the battery usage of each sensor node can be estimated such that the node is recharged well before its energy is depleted. When a node receives extra re-routed traffic after a failure in its neighborhood, however, the node may use up its energy faster than expectation and fail before the next battery recharge. As bringing a failed node back to service involves much overhead in reconfigurations, we assume that a failed node cannot be reactivated simply by a battery recharge. Therefore, given the battery recharge schedule, a node is subject to failure in each recharge cycle when it receives extra re-routed traffic. The failures are thus persistent because a node faces the risk of failure each time another node fails nearby.

In both failure scenarios, we observe a geographical constraint on the ability of failure to propagate, i.e., a failure has an impact radius. Nodes beyond the impact radius usually have no logical connection to the previous failure. For example, when a virus-infected node transmits the virus to other nodes, there exists a distance limit in one hop,

depending on the maximum available transmission power at the virus-infected node. In the traffic overloading example, similarly, the impact of a failed node is limited to the path repair region in which an alternative path segment is used to replace the broken one. In correspondence to this observation, we define the *failure impact radius* of node v_i as

$$r_i \triangleq \max_{\{v_j\}} \{d(v_i, v_j) : \mathrm{I}(v_i, v_j) = 1\}, \tag{5.1}$$

where $d(v_i, v_j) = \|X_i - X_j\|$ denotes the Euclidean distance between v_i and v_j, $\mathrm{I}(v_i, v_j) = 1$ if there is a causal connection between v_i and v_j, and $\mathrm{I}(v_i, v_j) = 0$ if there does not exist a causal connection between v_i and v_j. Considering the difference of the nodes in their respective failure impacts, for example nodes may have different transmission powers and different traffic loads, the impact radius r_i is modeled as a random variable with probability density function $f_{r_i}(x)$ $(0 \leq x \leq R)$. By the scaling property [112], we choose $R = 1$ without loss of generality. We also assume $r_1, r_2, \cdots, r_n$ are i.i.d. and unify their probability density functions as $f(x)$.

In addition, we notice in real-world scenarios that a node fails only probabilistically after its neighbor, even if the node is located within the failure impact radius of its neighbor. In the example of virus transmission, when node v_j is contacted by an infected node v_i, the result of infection or non-infection depends on whether v_j is vulnerable or not. In the example of traffic overloading, similarly, when v_j receives re-routed traffic from v_i, v_j may or may not fail, depending on the amount of re-routed traffic, the pre-existing traffic, the remaining energy, and the time to the next recharge. We hence define the *failure connection function* $g(x)$ $(0 \leq g(x) \leq 1)$ to model the likelihood of failure propagation from v_i to v_j. If $d(v_i, v_j) \leq r_i$, failure spreads to v_j with probability $g(d(v_i, v_j))$ that depends on their distance but not on their respective locations. If $d(v_i, v_j) > r_i$, failure cannot spread from v_i to v_j. For any two nodes v_{j_1} and v_{j_2}, the failure propagations from v_i to v_{j_1} and v_{j_2} are independent from each other.

To summarize our mathematical model for the characterization of failure spreading in wireless networks, we highlight our assumptions again. Our model defines three components to represent the node distribution in a wireless network and the failure correlation among nodes. The first component is the node density λ, which we assume to be a finite constant. The second component is the failure impact radius r, which we assume to be a random variable smaller than or equal to 1. The third component is the failure connection probability function $g(x)$, which we assume to be dependent on the distance between the

failed node and the normal node impacted by the failed node.

Note that according to our definitions, the failure impact radius and the failure connection probability of a given failed node are not related to the type of failure correlations. For both one-time and persistent failures, a failed node has the same probability distribution in its impact radius and the same probability function for its infection to another node. In the case of persistent failures, the persistence is observed from the perspective of a normal node as it might fail every time contacted by a different failed node. For a given failed node, however, its impact on a normal node is always one-time. At the time of a node failure, its effect on a normal node is instantaneous, which could be the attempt of transmitting a virus or the re-routing of traffic load. A given failed node does not contact a normal node multiple times. Therefore, we model the one-time impact of a given failed node on a given normal node by using the same impact radius and connection probability in both types of failures.

With different types of failure correlations, however, a normal node may fail eventually with different probabilities. In one-time failures, a normal node is at most subject to one failure risk. In persistent failures, a normal node is subject to one failure risk each time being contacted by a failed node. As the result, a normal node fails eventually with a higher probability in the type of persistent failures than in the type of one-time failures. Though a normal node is subject to multiple failure risks in the persistent failure type, we note that the ultimate failure probability does not converge to 1 because a normal node is within the impact radius of at most a constant number of failed nodes on average, since we assume a finite node density and a finite failure impact radius. The difference in the failure types however results in different failure spreading behaviors, which we will utilize the percolation theory to characterize in the next.

5.2.2 Preliminaries of Percolation

Percolation theory originated from many real-world connectivity phenomena. An example is liquid transfusion in a porous solid. The solid contains many tiny interstices (called sites) and neighbor interstices are connected by paths (called bonds). If one of the sites is wet, the liquid may reach other sites through the inter-connecting bonds. We say that a site or bond is *open* if the liquid can pass through it and *closed* otherwise.

There are two fundamental classes of percolation depending on the site locations.

In *discrete percolation* [113], the sites are assumed to be aligned in a regular lattice. Complementary to discrete percolation is *continuum percolation* [112] in which the sites are randomly located, usually assumed in a homogeneous Poisson distribution. Furthermore, if each site is open with some probability while all the bonds are always open, the percolation is also called *site percolation*. On the contrary, if each bond is open with probability while all the sites are always open, it is called *bond percolation*.

The essential problem studied in percolation is the component size. Starting from a given site, which is usually referred as the origin, we are interested in knowing how many other sites are eventually connected to the origin. The *percolation probability* is defined in the literature to measure the chance of existing an infinitely large component that contains the origin,

$$p_\infty \triangleq \Pr[|C_0| = \infty], \tag{5.2}$$

where $|C_0|$ denotes the size of the component containing the origin. When $|C_0| = \infty$, we call C_0 a *giant component*.

We are particularly interested in the existence of a giant component in the continuum percolation scenario. By assuming that a bond is open if and only if it connects two sites within distance 1, there exists a critical site density λ_c which is defined as

$$\lambda_c \triangleq \inf\{\lambda > 0 : p_\infty > 0\}, \tag{5.3}$$

such that $p_\infty > 0$ if $\lambda > \lambda_c$ and $p_\infty = 0$ if $\lambda < \lambda_c$. So far, the best known rigorous bounds on λ_c are $0.7698 < \lambda_c < 3.372$ [115]. Note that $p_\infty > 0$ indicates only the possibility for percolation, in addition to which the actual occurrence depends on the location of the origin. More specifically, it is almost sure that a giant component exists if $p_\infty > 0$, but the giant component may not contain the chosen origin.

5.2.3 Failure Percolation

Given the similarity between failure spread and liquid transfusion, we use the analytical techniques of percolation theory to characterize failure spread in large wireless networks. In percolation terminology, each node in the wireless network is a site and failure connections define the bonds between neighboring sites. Since node locations are randomly distributed and failure connections exist probabilistically, failure spread corresponds to the *random connection model* [112] of the continuum percolation scenario. When failure propagates from v_i to v_j in one hop, we say that the bond between v_i and v_j is open. Otherwise,

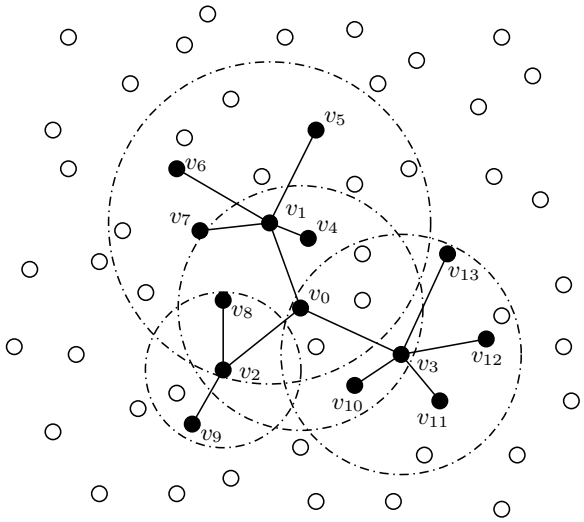


Figure 5.1: Failure spreading in large wireless networks.

the bond is closed. Note that the connections considered in this chapter represent the logical correlations among failures, which are completely different from the communication links.

An example of failure spread is illustrated in Fig. 5.1. In this example, the initial failure occurs at node v_0. As a result of this failure, nodes v_1–v_3 fail subsequently and spread the failure further away to nodes v_4–v_{13}. In each step of spread, a node that has just failed in the previous step passes failure to a random subset of nodes in its neighborhood, as modeled by the impact radius distribution function $f(x)$ and the connection function $g(x)$. As time goes, there are two possible results regarding failure spread: either the spread continues for ever or it stops automatically. Percolation theory tells us that failure does not percolate if the node density is very low. When node density is sufficiently high, percolation depends on functions $f(x)$ and $g(x)$. Our task is to characterize the failure correlations via $f(x)$ and $g(x)$ to determine the trend of failure spread given sufficiently high node density.

In the rest of this chapter, we will represent a wireless network as a *random geometric graph* [114] and denote it as $G(\mathcal{H}_\lambda, f(\cdot), g(\cdot))$. Accordingly, we denote the probability of failure percolation as $p_\infty(\lambda, f(\cdot), g(\cdot))$. As discussed, failure has the chance of percolation when $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$, but percolation does not necessarily happen. The occurrence depends on the location of the initial failure. For the purpose of concise presentation, however, we say that failure percolates whenever $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$. Besides, when percolation happens, failure does not necessarily reach every node. Rather, it reaches a significant portion of the network (at least a constant percentage). We say that failure spreads to the entire

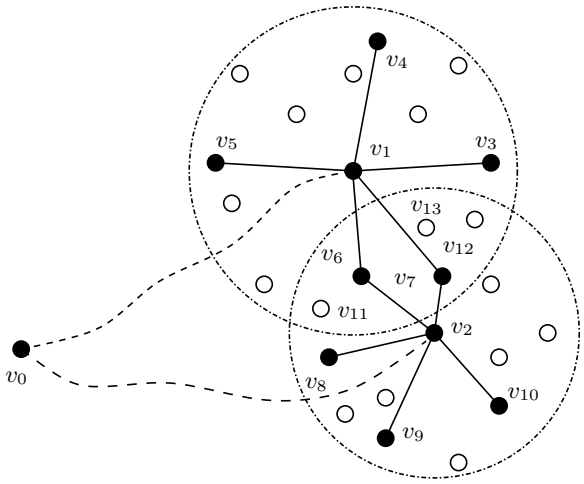


Figure 5.2: Spread of one-time failures.

network because it cannot be confined in any finite area when network size increases to infinity.

Our goal is to determine the respective regimes of functions $f(x)$ and $g(x)$ such that $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$ and $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$ with given λ. Obviously, $G(\mathcal{H}_\lambda, f(\cdot), g(\cdot))$ is a subgraph of $G(\mathcal{H}_\lambda, 1, 1)$ and $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$ whenever $p_\infty(\lambda, 1, 1) = 0$. To avoid triviality, we only consider the case $p_\infty(\lambda, 1, 1) > 0$ and determine the constraints on $f(x)$ and $g(x)$ for $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$ and $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$. Since $p_\infty(\lambda, 1, 1) > 0$ when $\lambda > \lambda_c$ and $0.7698 < \lambda_c < 3.372$, we assume $\lambda > 3.372$ in the rest of this chapter. In addition, if r is a constant, we also assume $\lambda r^2 > 3.372$, which guarantees $p_\infty(\lambda, r, 1) > 0$ by the scaling property [112].

5.3 Spread of One-Time Failures

In this section, we study the percolation of one possible category of failures. We say that the failures are one-time if each node is subject to the impact of other nodes only once. In reality, these failures happen due to a common security risk that is shared by a subset of nodes in the network.

We illustrate an example of one-time failures in Fig. 5.2, in which failure has occurred at v_0 and propagated via some paths to v_1 and v_2. Node v_1 spreads the failure to v_3–v_7, but not to v_{11}–v_{13}. Correspondingly, node v_2 must also spread the failure to v_6–v_7 but not to v_{11}–v_{13}, because v_6–v_7 are subject to this failure while v_{11}–v_{13} are immune.

For one-time failures, a node fails due to its own vulnerability, which is independent of the specific contact node and their distance. Therefore, we can simplify the failure connection function as $g(x) = p$, where p is a constant and $0 < p < 1$. Furthermore, $g(x)$ applies only to the first connection from a failed node. For subsequent connections, $g(x) = 0$ if the node did not fail at the first contact and $g(x) = 1$ if the node already failed. For clearer presentation, we denote the percolation probability of one-time failures as $p_\infty^{(\text{one})}(\lambda, f(\cdot), p)$, on which we have the following results.

5.3.1 Percolation with Constant Failure Impact Radius

First, we discuss the special case of a constant r for every node. In this case, we have the following theorem.

***Theorem* 5.1.** *For constant failure impact radius r,*

i) $p_\infty^{(\text{one})}(\lambda, r, p) > 0$ *if* $pr^2 > \frac{\lambda_c}{\lambda}$,

ii) $p_\infty^{(\text{one})}(\lambda, r, p) = 0$ *if* $pr^2 < \frac{\lambda_c}{\lambda}$.

Proof. Since every node is subject to the failure only once, regardless of which failed node contacts it first, we can predetermine the working status of each node with the failure probability p before the initial failure occurs. Obviously, percolation in the resulting graph of predetermined failed nodes has the same property as in the original graph in which each node may fail at the first contact from a failed node.

According to the Thinning Theorem [114], the predetermined failed nodes have density $p\lambda$ and $p_\infty^{(\text{one})}(\lambda, r, p) = p_\infty(p\lambda, r, 1)$. By the definition of λ_c in Eq. (5.3) and the scaling property [112], the critical density for $p_\infty(\lambda, r, 1) > 0$ is $\frac{\lambda_c}{r^2}$. Thus, $p_\infty(p\lambda, r, 1) > 0$ if $p\lambda > \frac{\lambda_c}{r^2}$ and $p_\infty(p\lambda, r, 1) = 0$ if $p\lambda < \frac{\lambda_c}{r^2}$. The results follow immediately. □

5.3.2 Percolation with Random Failure Impact Radius

When r is a random variable, the results in Theorem 5.1 can be generalized, as stated below in Theorem 5.2.

***Theorem* 5.2.** *For random failure impact radius r with probability density function $f(x)$ $(0 \le x \le 1)$,*

i) $p_\infty^{(\text{one})}(\lambda, f(\cdot), p) > 0$ *if* $ph_1(f(x)) > \frac{\lambda_c}{\lambda}$,

ii) $p_\infty^{(\text{one})}(\lambda, f(\cdot), p) = 0$ *if* $p(1 - h_2(f(x))) < \frac{\lambda_c}{\lambda}$,

where $h_1(f(x)) = \max_{\{0\le a\le 1\}}\{a^2 \int_a^1 f(x)\mathrm{d}x\}$, $h_2(f(x)) = \max_{\{0\le a\le 1\}}\{(1-a^2)\int_0^a f(x)\mathrm{d}x\}$.

Proof. Similar to the proof of Theorem 5.1, we have $p_\infty^{(\text{one})}(\lambda, f(\cdot), p) = p_\infty(p\lambda, f(\cdot), 1)$. We next focus on $p_\infty(p\lambda, f(\cdot), 1)$.

We first prove statement (i). When r has probability density function $f(x)$, given any $0 \le a \le 1$, $r \ge a$ with probability $\int_a^1 f(x)\mathrm{d}x$. We decompose graph $G(\mathcal{H}_{p\lambda}, f(\cdot), 1)$ into two subgraphs: one consisting of the nodes with $r \ge a$, denoted as $G(\mathcal{H}_{p\lambda\int_a^1 f(x)\mathrm{d}x}, f(\cdot), 1)$, and the other one consisting of the nodes with $r < a$, denoted as $G(\mathcal{H}_{p\lambda\int_0^a f(x)\mathrm{d}x}, f(\cdot), 1)$. Obviously, if failure percolates in $G(\mathcal{H}_{p\lambda\int_a^1 f(x)\mathrm{d}x}, f(\cdot), 1)$, it also percolates in $G(\mathcal{H}_{p\lambda}, f(\cdot), 1)$. Furthermore, every node in $G(\mathcal{H}_{p\lambda\int_a^1 f(x)\mathrm{d}x}, f(\cdot), 1)$ has $r \ge a$. If we reduce their r to the constant a and failure still percolates, then percolation must happen in $G(\mathcal{H}_{p\lambda\int_a^1 f(x)\mathrm{d}x}, f(\cdot), 1)$. Failure percolates in $G(\mathcal{H}_{p\lambda\int_a^1 f(x)\mathrm{d}x}, a, 1)$ (where $r = a$) when $p\lambda\int_a^1 f(x)\mathrm{d}x$ exceeds the critical density $\frac{\lambda_c}{a^2}$, i.e.,

$$pa^2 \int_a^1 f(x)\mathrm{d}x > \frac{\lambda_c}{\lambda}. \tag{5.4}$$

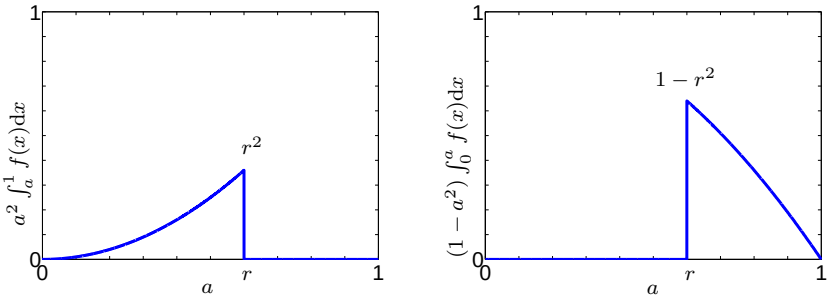
Finally, we note that percolation happens as long as we find any a that satisfies Eq. (5.4). In other words, condition

$$p \cdot \max_{\{0\le a\le 1\}} \Big\{a^2 \int_a^1 f(x)\mathrm{d}x\Big\} > \frac{\lambda_c}{\lambda} \tag{5.5}$$

suffices for $p_\infty(p\lambda, f(\cdot), 1) > 0$.

We next prove statement (ii). Similar to the graph decomposition method used above, we divide the original graph $G(\mathcal{H}_{p\lambda}, f(\cdot), 1)$ into the subgraph $G(\mathcal{H}_{p\lambda\int_a^1 f(x)\mathrm{d}x}, f(\cdot), 1)$ and the subgraph $G(\mathcal{H}_{p\lambda\int_0^a f(x)\mathrm{d}x}, f(\cdot), 1)$. For each node in $G(\mathcal{H}_{p\lambda\int_a^1 f(x)\mathrm{d}x}, f(\cdot), 1)$, we scale its impact radius r to ar. As a compensation for reducing r, we increase the node density from $p\lambda\int_a^1 f(x)\mathrm{d}x$ to $\frac{p\lambda}{a^2}\int_a^1 f(x)\mathrm{d}x$. The resulting subgraph $G(\mathcal{H}_{\frac{p\lambda}{a^2}\int_a^1 f(x)\mathrm{d}x}, f_a(\cdot), 1)$ has the same connectivity as the subgraph $G(\mathcal{H}_{p\lambda\int_a^1 f(x)\mathrm{d}x}, f(\cdot), 1)$. We then merge $G(\mathcal{H}_{\frac{p\lambda}{a^2}\int_a^1 f(x)\mathrm{d}x}$, $f_a(\cdot), 1)$ and $G(\mathcal{H}_{p\lambda\int_0^a f(x)\mathrm{d}x}, f(\cdot), 1)$ into a graph $G(\mathcal{H}_{\lambda'}, f'(\cdot), 1)$ with $\lambda' = p\lambda(\int_0^a f(x)\mathrm{d}x + \frac{1}{a^2}\int_a^1 f(x)\mathrm{d}x)$ and $r' \le a$, which has the same percolation property as $G(\mathcal{H}_{p\lambda}, f(\cdot), 1)$. When $\lambda' < \frac{\lambda_c}{a^2}$, i.e.,

$$p\lambda\Big(\int_0^a f(x)\mathrm{d}x + \frac{1}{a^2}\int_a^1 f(x)\mathrm{d}x\Big) < \frac{\lambda_c}{a^2}, \tag{5.6}$$



(a) $a^2 \int_a^1 f(x)\mathrm{d}x$ (b) $(1-a^2) \int_0^a f(x)\mathrm{d}x$

Figure 5.3: Functions $a^2 \int_a^1 f(x)\mathrm{d}x$ and $(1-a^2) \int_0^a f(x)\mathrm{d}x$ when r is a constant, i.e., $f(x) = \delta(x-r)$.

failure does not percolate in $G(\mathcal{H}_{\lambda'}, a, 1)$, and thus not in $G(\mathcal{H}_{\lambda'}, f'(\cdot), 1)$ and $G(\mathcal{H}_{p\lambda}, f(\cdot), 1)$. Rewriting Eq. (5.6) to

$$p\Big(1-(1-a^2)\int_0^a f(x)\mathrm{d}x\Big) < \frac{\lambda_c}{\lambda}, \tag{5.7}$$

and noting that percolation does not happen as long as there exists any a that satisfies Eq. (5.7), we arrive at

$$p\Big(1-\max_{\{0\le a\le 1\}}\Big\{(1-a^2)\int_0^a f(x)\mathrm{d}x\Big\}\Big) < \frac{\lambda_c}{\lambda}, \tag{5.8}$$

which suffices for $p_\infty(p\lambda, f(\cdot), 1) = 0$. □

Note that Theorem 5.1 is in fact a special case of Theorem 5.2. When r is a constant instead of a random variable, $f(x) = \delta(x-r)$, where $\delta(x-r)$ is called the *Dirac delta function* with the properties: (i) $\delta(x-r) = \infty$ when $x = r$ and $\delta(x-r) = 0$ when $x \neq r$, and (ii) $\int_{r_-}^{r_+} \delta(x-r)\mathrm{d}x = 1$. We depict functions $a^2 \int_a^1 f(x)\mathrm{d}x$ and $(1-a^2) \int_0^a f(x)\mathrm{d}x$ in Fig. 5.3 with $f(x) = \delta(x-r)$. It is observed that these two functions achieve their maxima at $a = r$. Therefore, we have $h_1(f(x)) = r^2$ and $h_2(f(x)) = 1-r^2$. Plugging them into Theorem 5.2, we reach the same results as in Theorem 5.1.

Intuitively, r and p represent the degree of failure contagiousness. When r and p increase, failure tends to percolate. When they decrease, percolation becomes unlikely to happen. Our results in Theorems 5.1 and 5.2 present a quantified measure on r and p. Moreover, in the general case of a random r, Theorem 5.2 indicates that the chance of failure percolation increases if the probability distribution $f(x)$ shifts toward large r (such that $h_1(f(x))$ increases) and decreases if $f(x)$ shifts toward small r (such that $h_2(f(x))$ increases).

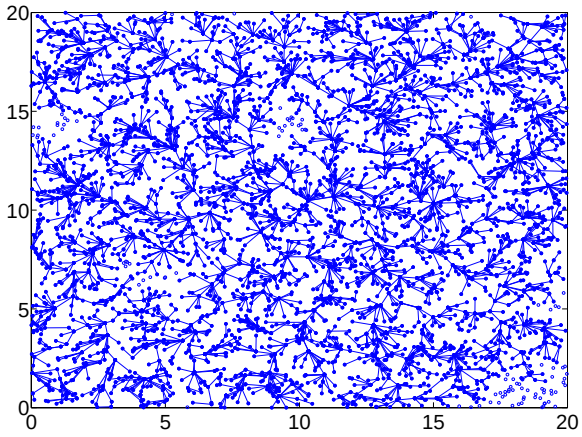


Figure 5.4: Failure percolation with $p\lambda = 9.8$ and uniformly distributed $f(x)$.

5.3.3 Simulation Results

As a visual verification of Theorem 5.2, we show our simulation results on failure percolation in Fig. 5.4 and Fig. 5.5. Because Theorem 5.1 is a special case of Theorem 5.2, we do not provide separate simulation results for Theorem 5.1.

In our simulations, we assume that $f(x)$ is uniformly distributed in the range $0 \leq x \leq 1$. It is not difficult to find $h_1(f(x)) = 0.1481$ and $h_2(f(x)) = 0.3849$. By Theorem 5.2, we know that failure percolates if $0.1481p\lambda > \lambda_c$ and does not percolate if $0.6151p\lambda < \lambda_c$. Given $0.7698 < \lambda_c < 3.372$, we need to verify the percolation results for $p\lambda > \frac{3.372}{0.1481} = 22.7684$ and $p\lambda < \frac{0.7698}{0.6151} = 1.2515$ respectively.

However, as the existing simulation result has demonstrated that $1.43 < \lambda_c < 1.44$ with high confidence [115], we simulate $p\lambda > \frac{1.44}{0.1481} = 9.7232$ and $p\lambda < \frac{1.43}{0.6151} = 2.3248$ instead. If failure percolates with $p\lambda > 9.7232$, it must percolate with $p\lambda > 22.7684$. Similarly, if failure does not percolate with $p\lambda < 2.3248$, it does not with $p\lambda < 1.2515$ either.

We choose $p\lambda = 9.8$ and $p\lambda = 2.3$ respectively, and present the results of failure spread in a 20×20 area in Fig. 5.4 and Fig. 5.5, where the initial failure occurs at the center of this area. For clearer presentation, we have only shown in the figures the connections through which failure is propagated from a failed node to a normal node (which becomes failed after the propagation) but ignored the connections between any two nodes that have already failed before these connections are used for failure propagation. We observe that

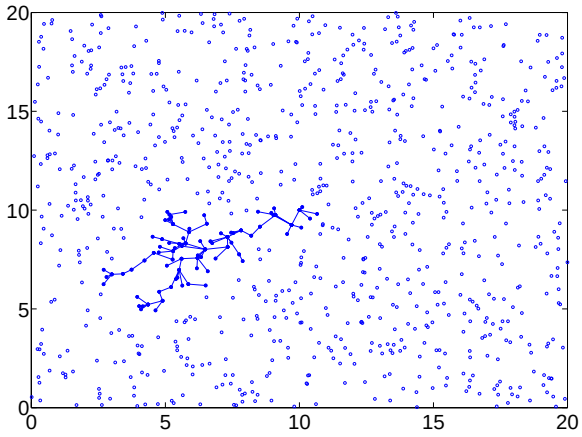


Figure 5.5: Failure percolation with $p\lambda = 2.3$ and uniformly distributed $f(x)$.

failure is able to spread to a majority of nodes in Fig. 5.4 while only to a limited small cluster of nodes in Fig. 5.5, hence supporting our claims in Theorem 5.2.

5.4 Spread of Persistent Failures: Constant Failure Correlations

Another widely observed category of failures is that a node may be impacted by every failure that occurs in its neighborhood. Compared to one-time failures, persistent failures are easier to percolate in the network. Intuitively, a node subject to failure at each contact from failed nodes has an increased chance to join the cluster of failed nodes and, therefore, the giant component becomes likely to emerge.

An example of persistent failures is given in Fig. 5.6, in which the initial failure at v_0 has spread to v_1 and v_2. Nodes v_1 and v_2 cause further failures at nodes v_3–v_7 and v_8–v_{12} respectively. Note that v_2 does not need to fail v_6–v_7 again (though this could happen) since v_6–v_7 are already part of the failed component connected through v_1. However, v_2 may fail v_{11}–v_{12}, which have withstood the impact of v_1. In comparison to Fig. 5.2, the component of failed nodes is larger in Fig. 5.6.

We explore the percolation of persistent failures in this and next sections. Specifically, we focus on constant failure impact radius r and constant failure connection probability p in this section. We will study the generalized functions $f(x)$ and $g(x)$ in the

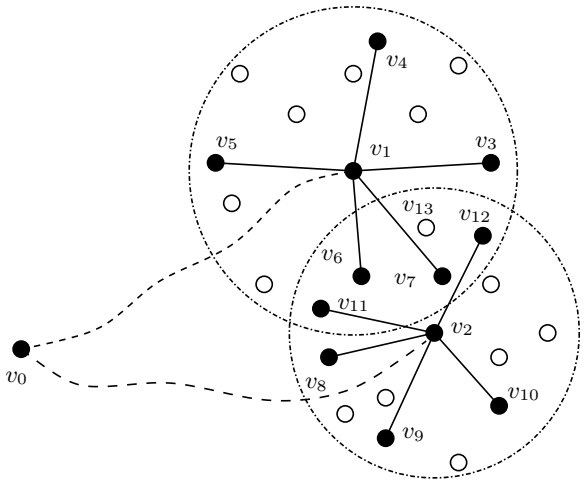


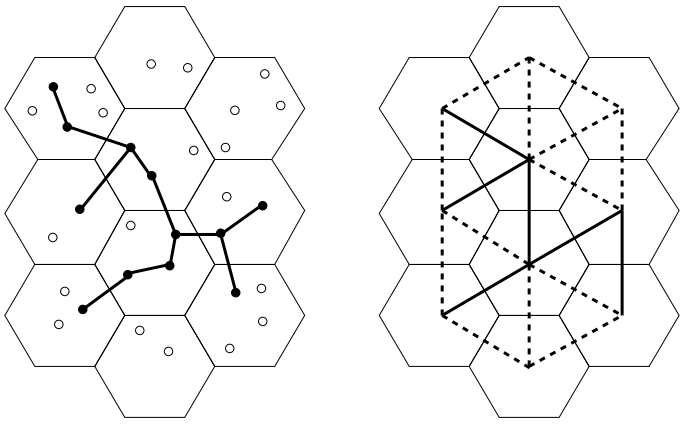
Figure 5.6: Spread of persistent failures.

following section. Next, we discuss separately the sufficient conditions for percolation and non-percolation of persistent failures with constant r and p. We use $p_\infty(\lambda, r, p)$ in this section to denote the percolation probability of persistent failures with constant r and p.

5.4.1 Sufficient Condition of Percolation

Our discussion on the difference between persistent and one-time failures shows that persistent failures are easier to percolate. Therefore, percolation of persistent failures should happen under the same sufficient condition for one-time failures, i.e., $pr^2 > \frac{\lambda_c}{\lambda}$. As the current best known rigorous bounds on λ_c are $0.7698 < \lambda_c < 3.372$, we infer that percolation occurs when $pr^2 > \frac{3.372}{\lambda}$. In regime $pr^2 < \frac{3.372}{\lambda}$, we have no idea on failure percolation. Further judgment depends on the improved accuracy of the bounds on λ_c.

However, since persistent failures are easier to percolate than one-time failures, we expect a tighter sufficient condition that allows percolation with smaller r and p than $pr^2 > \frac{3.372}{\lambda}$. We next determine a new condition by using the technique of continuum-to-discrete percolation mapping. We divide the network area into many small hexagonal cells, as illustrated in Fig. 5.7(a). The graph of the failed nodes and their connections now appears on the background of these cells. As failure spreads, it travels through a cluster of continuous cells. We define a cell as *open* if it contains at least one failed node and *closed* otherwise. Let C_0^{cell} denote the cluster of open cells and $|C_0^{\text{cell}}|$ denote its size. It is obviously true that if $|C_0| = \infty$, then $|C_0^{\text{cell}}| = \infty$, and vice versa. The mapping between the cluster



(a) continuum (b) discrete

Figure 5.7: Mapping from continuum percolation to discrete percolation.

of failed nodes and the cluster of open cells thus allows us to find the sufficient condition for $|C_0^{\text{cell}}| = \infty$ and use it for $|C_0| = \infty$.

The cluster size of open cells is studied via bond percolation on a discrete lattice, as shown in Fig. 5.7(b). In this abstract form, we represent a cell by a site located at the center of the cell. Two neighboring sites are connected by a bond, which represents the connectivity between the two corresponding cells. We define the bond to be *open* if, given one or more failed nodes in one of the end cells, at least one failed node connects to some nodes in the other end cell. In other words, an open bond propagates failure from one cell to another. In Fig. 5.7(b), open bonds are denoted as solid lines that are consistent with the continuum percolation in Fig. 5.7(a).

We observe in Fig. 5.7(b) that when the cells have hexagonal shape, the discrete lattice is triangular. In the literature, square cells are usually used [116,117] as they generate square lattice that is simple and easy for analysis. We use hexagonal cells for two reasons. First, hexagonal cells yield a tighter bound on r and p for failure percolation than square cells. Second, they render the study of failure non-percolation possible under this mapping framework. With triangular lattice, there exists a critical probability $p_c^{\Delta} = 2\sin(\frac{\pi}{18}) = 0.3473$ [113] such that percolation occurs, i.e., $|C_0^{\text{cell}}| = \infty$, if each bond is open with a probability higher than p_c^{Δ} and not occurs, i.e., $|C_0^{\text{cell}}| < \infty$, otherwise. The discrete percolation in triangular lattice allows us to derive a tighter sufficient condition for failure percolation. Before presenting our result, we introduce two lemmas first, which will be used

in the subsequent proof.

***Lemma* 5.1.** *Given a failed node v_i in cell c_i and a neighboring cell c_j, if none of the nodes in c_j is connected to v_i, then the probability of existing k nodes in c_j is $P(k) = e^{-(1-p)\lambda s}\frac{((1-p)\lambda s)^k}{k!}$, where p is the failure connection probability, λ is the node density, s is the area of a cell.*

Proof. Let E_1 denote the event that there exist k nodes in c_j and E_2 denote the event that none of the nodes in c_j is connected to v_i. Then the conditional probability is written as $P(k) = \frac{\Pr[E_1 \cap E_2]}{\Pr[E_2]}$, from which we have

$$\begin{aligned} P(k) &= \frac{e^{-\lambda s}\frac{(\lambda s)^k}{k!}(1-p)^k}{e^{-p\lambda s}} \\ &= e^{-(1-p)\lambda s}\frac{((1-p)\lambda s)^k}{k!}, \end{aligned} \tag{5.9}$$

which shows that the conditional node distribution is Poisson with density $(1-p)\lambda$. □

***Lemma* 5.2.** *Given neighboring cells c_i and c_j, assume that the number of failed nodes in c_i is Poisson distributed with density λ_i and the number of nodes in c_j is Poisson distributed with density λ_j. The probability of failure spread from c_i to c_j is bounded as $P \geq (1 - e^{-\lambda_i s})(1 - e^{-p\lambda_j s})$, where p is the failure connection probability, s is the area of a cell.*

Proof. As long as there exists one open connection between a failed node in c_i and a node in c_j, failure propagates from c_i to c_j. The probability is then expressed as

$$\begin{aligned} P &= \sum_{k_1=0}^{\infty}\sum_{k_2=0}^{\infty} e^{-\lambda_i s}\frac{(\lambda_i s)^{k_1}}{k_1!}e^{-\lambda_j s}\frac{(\lambda_j s)^{k_2}}{k_2!}(1-(1-p)^{k_1 k_2}) \\ &= 1 - \sum_{k_1=0}^{\infty}\sum_{k_2=0}^{\infty} e^{-\lambda_i s}\frac{(\lambda_i s)^{k_1}}{k_1!}e^{-\lambda_j s}\frac{(\lambda_j s)^{k_2}}{k_2!}(1-p)^{k_1 k_2} \\ &= 1 - \sum_{k_1=0}^{\infty} e^{-\lambda_i s}\frac{(\lambda_i s)^{k_1}}{k_1!}\sum_{k_2=0}^{\infty} e^{-\lambda_j s}\frac{(\lambda_j s)^{k_2}}{k_2!}(1-p)^{k_1 k_2} \\ &= 1 - \sum_{k_1=0}^{\infty} e^{-\lambda_i s}\frac{(\lambda_i s)^{k_1}}{k_1!}e^{-\lambda_j s(1-(1-p)^{k_1})} \\ &= 1 - e^{-\lambda_i s} - \sum_{k_1=1}^{\infty} e^{-\lambda_i s}\frac{(\lambda_i s)^{k_1}}{k_1!}e^{-\lambda_j s(1-(1-p)^{k_1})} \end{aligned}$$

$$\begin{aligned} &\geq 1 - e^{-\lambda_i s} - \sum_{k_1=1}^{\infty} e^{-\lambda_i s} \frac{(\lambda_i s)^{k_1}}{k_1!} e^{-p\lambda_j s} \\ &= 1 - e^{-\lambda_i s} - e^{-p\lambda_j s}(1 - e^{-\lambda_i s}) \\ &= (1 - e^{-\lambda_i s})(1 - e^{-p\lambda_j s}), \end{aligned} \tag{5.10}$$

where the inequality occurs because $1 - (1-p)^{k_1} \geq p$ when $k_1 \geq 1$. □

Based on Lemmas 5.1 and 5.2, we have the following theorem.

***Theorem* 5.3.** *For constant impact radius r and connection probability p, $p_\infty(\lambda, r, p) > 0$ if $pr^2 > \frac{1.8889}{\lambda}$.*

Proof. We choose the size of each cell small enough such that given two arbitrary locations in two neighboring cells, one in each, their distance is at most r. This small cell size guarantees that a failed node has a positive chance to contact and infect every node in the neighboring cells. Given hexagonal cells with radius l, it is not difficult to find the maximum distance between two nodes in neighboring cells as $l\sqrt{13}$. Letting $l\sqrt{13} = r$, we obtain $l = \frac{r}{\sqrt{13}}$ and $s = \frac{3\sqrt{3}}{2}l^2 = \frac{3\sqrt{3}}{26}r^2$, where s is the area of each cell. Note that with this small cell size, it is possible for a failed node to connect in one hop to the nodes in non-neighboring cells. However, we can safely ignore these "jumping" connections, because they increase the probability of failure percolation and therefore do not change the validity of our result in this theorem.

We use p^{bond} to denote the bond open probability in the triangular lattice. We next derive a bound for p^{bond} by considering two possible ways for failure to spread from a node v_i in cell c_i to a neighboring cell c_j: (i) direct connections from v_i to some nodes in c_j, and (ii) connections from v_i to stepstones in c_i and connections from these stepstones to some nodes in c_j.

In the first case, v_i connects to each node in the neighboring cell c_j with probability p independently. As the number of nodes in c_j follows a Poisson distribution with density λ, the number of nodes in c_j that fail after v_i is also a Poisson random variable, with density $p\lambda$. Hence, the probability of at least one node in c_j being connected by v_i is $P_1 = 1 - e^{-p\lambda s}$.

With the complementary probability $P_1^c = e^{-p\lambda s}$, there are no direct connections, in which we consider the stepstones. As nodes are Poisson distributed with density λ, the number of nodes neighboring with v_i inside cell c_i is also Poisson with density λ. The appearance of v_i does not change the distribution of its neighbors. For each of these neighbors,

v_i has a chance of p to connect to it independently from other neighbors. Therefore, the number of stepstones that v_i has is a Poisson random variable with density $p\lambda$. We next consider the node distribution in cell c_j. By Lemma 5.1, under the condition that none of the nodes in c_j is connected to v_i, the number of nodes in c_j is a Poisson random variable with density $(1-p)\lambda$. Letting $\lambda_i = p\lambda$ and $\lambda_j = (1-p)\lambda$, by Lemma 5.2, we obtain the probability of existing connections via stepstones when there are no direct connections from node v_i to cell c_j as $P_2 \geq (1-e^{-p\lambda s})(1-e^{-p(1-p)\lambda s})$.

Combining both cases and also considering the possibility of existing other failed nodes in c_i besides v_i, we bound the probability of failure propagation from c_i to c_j as

$$\begin{aligned} p^{\text{bond}} &\geq P_1 + P_1^c P_2 \\ &\geq 1 - e^{-p\lambda s} + e^{-p\lambda s}(1-e^{-p\lambda s})(1-e^{-p(1-p)\lambda s}) \\ &= (1-e^{-p\lambda s})(1+e^{-p\lambda s} - e^{-p(2-p)\lambda s}) \end{aligned} \tag{5.11}$$

By defining $y = p\lambda s$ and noting $\lambda s = \frac{3\sqrt{3}}{26}\lambda r^2 > \frac{3\sqrt{3}}{26} \times 3.372 = 0.6739$, we rewrite Eq. (5.11) as

$$\begin{aligned} p^{\text{bond}} &\geq (1-e^{-y})(1+e^{-y} - e^{-y(2-\frac{y}{\lambda s})}) \\ &> (1-e^{-y})(1+e^{-y} - e^{-y(2-\frac{y}{0.6739})}). \end{aligned} \tag{5.12}$$

Now we need to determine the condition on y such that

$$(1-e^{-y})(1+e^{-y} - e^{-y(2-\frac{y}{0.6739})}) > p_c^{\triangle} = 0.3473. \tag{5.13}$$

There is no closed-form solution for Eq. (5.13). Computing numerically, we find that Eq. (5.13) holds when $y > 0.3775$. Since $y = p\lambda s$ and $s = \frac{3\sqrt{3}}{26}r^2$, we arrive at

$$\frac{3\sqrt{3}}{26} p\lambda r^2 > 0.3775, \tag{5.14}$$

and finally

$$pr^2 > \frac{1.8889}{\lambda}, \tag{5.15}$$

under which $\Pr[|C_0^{\text{cell}}| = \infty] > 0$ and $p_\infty(\lambda, r, p) > 0$. □

5.4.2 Sufficient Condition of Non-Percolation

Our result on the sufficient condition of percolation have quantified r and p that are large enough for an initial failure to spread to the entire network. Similarly, we are also

interested in determination of r and p when failure cannot percolate in the network. From the continuum-to-discrete percolation mapping, we know that $|C_0| < \infty$ if $|C_0^{\text{cell}}| < \infty$. Thus, our task is to find the condition for $|C_0^{\text{cell}}| < \infty$.

We still divide the network into many hexagonal cells. Compared to square cells, failure non-percolation is easier to study with hexagonal cells. When we choose the cell size sufficiently large, the failed nodes in one cell can only connect to the six neighboring cells and the connections are symmetric in probability. We can focus on the connections between any two neighboring cells to understand failure spreading. With square cells, however, a cell has eight neighbors: four horizontal or vertical ones and four diagonal ones. Its connections to the horizontal or vertical neighbors are not symmetric to those to the diagonal neighbors, rendering failure percolation difficult to analyze. By mapping into hexagonal cells, we have the following theorem to characterize the sufficient condition on r and p for failure non-percolation.

***Theorem* 5.4.** *For constant impact radius r and connection probability p, $p_\infty(\lambda, r, p) = 0$ if $pr^2 < \frac{-\ln(1-\frac{0.1642}{\lambda})}{2.5981\lambda}$.*

Proof. We choose the cell radius $l = r$ such that a failed node in a cell can only connect to the six neighboring cells. In this way, the connections correspond to the triangular lattice illustrated in Fig. 5.7(b) and there are no jumping connections, i.e., a failed node cannot connect to non-neighboring cells in one hop. With $l = r$, the area of each cell is $s = \frac{3\sqrt{3}}{2}r^2$.

We examine the failure propagation from cell c_i to a neighboring cell c_j. Since there exist failed nodes in c_i, i.e., c_i is not empty, the number of nodes k_1 in c_i is a random variable slightly different from Poisson, given by $P(k_1) = \frac{e^{-\lambda s}(\lambda s)^{k_1}}{(1-e^{-\lambda s})k_1!}$ $(k_1 \geq 1)$. The number of nodes k_2 in c_j is still Poisson with density λ. Because the failed nodes are a subset of k_1 and a failed node may not be able to contact every node in k_2 due to the limit of r, we bound the probability of failure propagation from c_i to c_j as

$$\begin{aligned} p^{\text{bond}} &\leq \sum_{k_1=1}^{\infty}\sum_{k_2=0}^{\infty} \frac{e^{-\lambda s}(\lambda s)^{k_1}}{(1-e^{-\lambda s})k_1!}\frac{e^{-\lambda s}(\lambda s)^{k_2}}{k_2!}(1-(1-p)^{k_1 k_2}) \\ &= 1 - \sum_{k_1=1}^{\infty} \frac{e^{-\lambda s}(\lambda s)^{k_1}}{(1-e^{-\lambda s})k_1!} e^{-\lambda s(1-(1-p)^{k_1})} \\ &\leq 1 - \sum_{k_1=1}^{\infty} \frac{e^{-\lambda s}(\lambda s)^{k_1}}{(1-e^{-\lambda s})k_1!} e^{-k_1 p\lambda s} \end{aligned}$$

$$= 1 - \frac{e^{-\lambda s}}{(1-e^{-\lambda s})}(e^{\lambda s e^{-p\lambda s}} - 1)$$
$$= \frac{1 - e^{-\lambda s(1-e^{-p\lambda s})}}{1 - e^{-\lambda s}}. \quad (5.16)$$

Noting that $\lambda s = \frac{3\sqrt{3}}{2}\lambda r^2 > \frac{3\sqrt{3}}{2} \times 3.372 = 8.7607$ and $s = \frac{3\sqrt{3}}{2}r^2 \leq \frac{3\sqrt{3}}{2} = 2.5981$, we rewrite Eq. (5.16) as

$$p^{\text{bond}} < \frac{1 - e^{-2.5981\lambda(1-e^{-p\lambda s})}}{1 - e^{-8.7607}}. \quad (5.17)$$

By solving

$$\frac{1 - e^{-2.5981\lambda(1-e^{-p\lambda s})}}{1 - e^{-8.7607}} < p_c^{\Delta} = 0.3473, \quad (5.18)$$

we finally obtain

$$pr^2 < \frac{-\ln(1 - \frac{0.1642}{\lambda})}{2.5981\lambda} \quad (5.19)$$

for $\Pr[|C_0^{\text{cell}}| = \infty] = 0$ and $p_\infty(\lambda, r, p) = 0$. □

5.4.3 Simulation Results

We have simulated the spread of persistent failures to verify our results in Theorems 5.3 and 5.4. We generate Poisson distributed nodes with density $\lambda = 5$ in a 20×20 area. The initial failure occurs at the center of this area. We set the failure impact radius to be constant $r = 1$ and the failure connection probability to be constant $p = 0.38$ and $p = 0.0025$ respectively. The configurations of λ, r and p satisfy the conditions in Theorems 5.3 and 5.4 respectively. The simulation result for $p = 0.38$ is shown in Fig. 5.8, in which we observe that the failure spreads to most of the nodes in the network. On the contrary we observe in Fig. 5.9 for $p = 0.0025$ that the failure spreads to very limited few nodes only.

5.5 Spread of Persistent Failures: General Failure Correlations

When the failure impact radius r is random and the failure connection function $g(x)$ is in a general form, modeling of failure spread becomes even complicated. We present two theorems in this section that characterize the general failure correlations such that an initial failure will and will not percolate in the network. To facilitate our study, we first combine the functions $f(x)$ and $g(x)$ into a unified failure connection function, as stated in the lemma below.

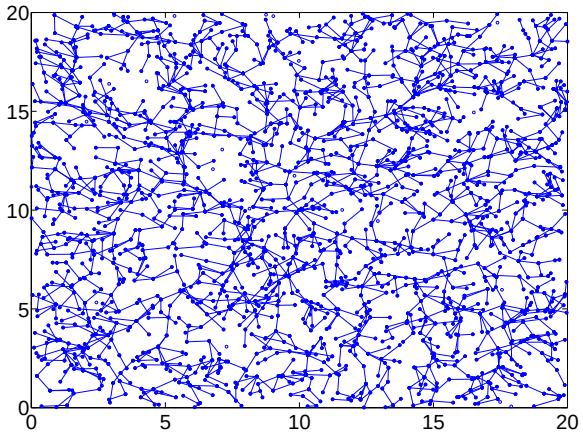


Figure 5.8: Failure percolation with $\lambda = 5$, $r = 1$, and $p = 0.38$.

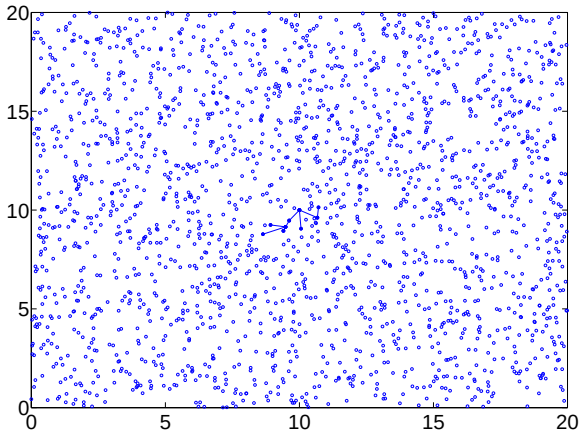


Figure 5.9: Failure percolation with $\lambda = 5$, $r = 1$, and $p = 0.0025$.

***Lemma* 5.3.** *For random failure impact radius r with probability density function $f(x)$ ($0 \leq x \leq 1$) and general failure connection function $g(x)$, the failure correlations can be modeled equivalently by a constant $r' = 1$ and a unified connection function $g'(x) = g(x) \int_x^1 f(z)\mathrm{d}z$ if $0 \leq x \leq 1$ and $g'(x) = 0$ if $x > 1$.*

Proof. When r is random and $g(x)$ is general, the connection between a failed node v_i and another node v_j is open with probability $g(x)$ conditional on $r_i \geq x$ and closed otherwise, where $x = \|X_i - X_j\|$ and r_i is the impact radius of v_i. Hence, the unconditional probability of an open connection is $g'(x) = g(x) \int_x^1 f(z)\mathrm{d}z$ if $0 \leq x \leq 1$. If $x > 1$, the connection is

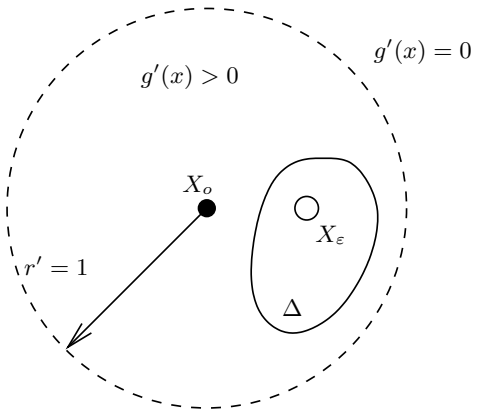


Figure 5.10: Function $\Psi(g'(x), X_o, \Delta) \triangleq \int_\Delta g'(\|X_o - X_\varepsilon\|)\mathrm{d}\varepsilon$.

closed because $r_i \leq 1 < x$, so $g'(x) = 0$. Since $g'(x)$ has considered both $f(x)$ and $g(x)$, we can always assume the failure impact radius as a constant $r' = 1$. □

Given the equivalence in modeling failure correlations, we consider r' and $g'(x)$ instead of $f(x)$ and $g(x)$ in the rest of this section. For clearer presentation, we now define a few concepts that will be used in our following theorems. We define function

$$\Psi(g'(x), X_o, \Delta) \triangleq \int_\Delta g'(\|X_o - X_\varepsilon\|)\mathrm{d}\varepsilon, \tag{5.20}$$

which is the integration of the probability $g'(x)$ over region Δ with respect to location X_o, as illustrated in Fig. 5.10. In addition, with respect to all the possible X_o and Δ ($|\Delta| = \sigma$), we define $\Psi_{\min}(g'(x), \sigma) = \min_{\{X_o, |\Delta|=\sigma\}}\{\Psi(g'(x), X_o, \Delta)\}$ and $\Psi_{\max}(g'(x), \sigma) = \max_{\{X_o, |\Delta|=\sigma\}}\{\Psi(g'(x), X_o, \Delta)\}$, which denote the minimum and maximum of $\Psi(g'(x), X_o, \Delta)$ respectively when we change X_o and Δ arbitrarily, as long as the area of Δ is kept constantly as σ. With the help of these definitions, we present our results regarding failure percolation with general $f(x)$ and $g(x)$ as follows.

5.5.1 Sufficient Condition of Percolation

***Theorem* 5.5.** *For general failure correlations $f(x)$ and $g(x)$, $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$ if $\Psi_{\min}(g'(x), 0.1999) > \frac{0.4266}{\lambda}$.*

Proof. We follow the same line of reasoning as in the proof of Theorem 5.3, but only consider case (i). Consideration of both cases (i) and (ii) will render this analysis intractable. Substituting $r' = 1$ for r, we obtain the area of each cell as $s = \frac{3\sqrt{3}}{26} = 0.1999$. Given a

failed node v_i in cell c_i, because each node v_j in cell c_j is connected to v_i independently with probability $g'(\|X_i - X_j\|)$, the number of open connections is a Poisson random variable with parameter $\lambda\Psi(g'(x), X_i, c_j)$. Since there may exist other failed nodes in c_i besides v_i that also connect to c_j, we bound the probability of failure propagation from c_i to c_j as

$$\begin{aligned} p^{\text{bond}} &\geq 1 - e^{-\lambda\Psi(g'(x),X_i,c_j)} \\ &\geq 1 - e^{-\lambda\Psi_{\min}(g'(x),0.1999)}. \end{aligned} \tag{5.21}$$

By solving $1 - e^{-\lambda\Psi_{\min}(g'(x),0.1999)} > p_c^{\triangle} = 0.3473$, we obtain

$$\Psi_{\min}(g'(x), 0.1999) > \frac{0.4266}{\lambda} \tag{5.22}$$

for $\Pr[|C_0^{\text{cell}}| = \infty] > 0$ and $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$. □

5.5.2 Sufficient Condition of Non-Percolation

***Theorem* 5.6.** *For general failure correlations $f(x)$ and $g(x)$, $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$ if $\Psi_{\max}($ $g'(x), 1.2535) < \frac{-\ln(1-\frac{0.1642}{\lambda})}{\lambda}$.*

Proof. We follow the same line of reasoning as in the proof of Theorem 5.4. Substituting $r' = 1$ for r, we obtain the area of each cell as $s = \frac{3\sqrt{3}}{2} = 2.5981$. Given $r' = 1$, a failed node v_i in cell c_i cannot reach all the locations in c_j, but only a fraction of area 1.2535 at maximum. When there exist k_1 number of nodes in c_i, denoted as $\{v_{i_1}, \cdots, v_{i_{k_1}}\}$, because the failed nodes are a subset of $\{v_{i_1}, \cdots, v_{i_{k_1}}\}$ and each failed node propagates failure to a node v_j in cell c_j independently, v_j fails after $\{v_{i_1}, \cdots, v_{i_{k_1}}\}$ with a probability at most $1-\prod_{m=1}^{k_1}(1-g'(\|X_{i_m}-X_j\|))$. Hence, the number of nodes that fail in c_j after $\{v_{i_1}, \cdots, v_{i_{k_1}}\}$ is a Poisson random variable with parameter Λ that satisfies

$$\begin{aligned} \Lambda &\leq \lambda \int_{c_j} 1 - \prod_{m=1}^{k_1} (1 - g'(\|X_{i_m} - X_\varepsilon\|)) \mathrm{d}\varepsilon \\ &\leq \lambda \int_{c_j} \sum_{m=1}^{k_1} g'(\|X_{i_m} - X_\varepsilon\|) \mathrm{d}\varepsilon \\ &= \lambda \sum_{m=1}^{k_1} \Psi(g'(x), X_{i_m}, c_j) \\ &\leq k_1 \lambda \Psi_{\max}(g'(x), 1.2535). \end{aligned} \tag{5.23}$$

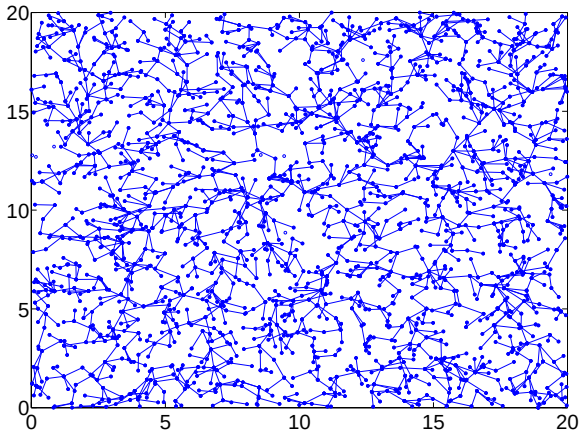


Figure 5.11: Failure percolation with $\lambda = 5$, $r' = 1$, and $g'(x) = 1 - 0.56x$.

Similar to Eq. (5.16), by considering the distribution of k_1,

$$\begin{aligned} p^{\text{bond}} &= \sum_{k_1=1}^{\infty} \frac{e^{-\lambda s}(\lambda s)^{k_1}}{(1-e^{-\lambda s})k_1!}(1-e^{-\Lambda}) \\ &\leq 1 - \sum_{k_1=1}^{\infty} \frac{e^{-\lambda s}(\lambda s)^{k_1}}{(1-e^{-\lambda s})k_1!} e^{-k_1 \lambda \Psi_{\max}(g'(x),1.2535)} \\ &= \frac{1-e^{-\lambda s(1-e^{-\lambda \Psi_{\max}(g'(x),1.2535)})}}{1-e^{-\lambda s}}. \end{aligned} \tag{5.24}$$

Furthermore, by following Eq. (5.17), (5.18) and (5.19), we obtain

$$\Psi_{\max}(g'(x), 1.2535) < \frac{-\ln(1-\frac{0.1642}{\lambda})}{\lambda} \tag{5.25}$$

for $\Pr[|C_0^{\text{cell}}| = \infty] = 0$ and $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$. □

5.5.3 Simulation Results

To verify our analytical results in Theorems 5.5 and 5.6, we consider a special case of the function $g'(x)$. We let $g'(x)$ be a linearly decreasing function of x, i.e., the probability of failure propagation decreases as the hop distance increases. As before, we generate randomly located nodes in a 20×20 area with node density $\lambda = 5$. The initial failure happens at the center of this area. We first specify $g'(x) = 1 - 0.56x$ $(0 \leq x \leq 1)$, which satisfies the condition given in Theorem 5.5. The simulation result is shown in Fig. 5.11. It is observed that the failure percolates in the network. We then require $g'(x) = 0.0052-0.0052x$

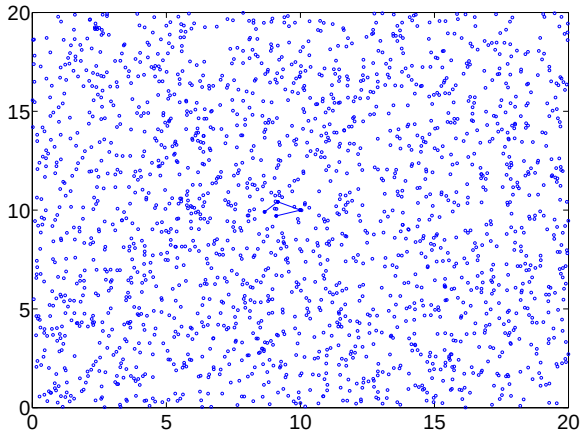


Figure 5.12: Failure percolation with $\lambda = 5$, $r' = 1$, and $g'(x) = 0.0052 - 0.0052x$.

$(0 \leq x \leq 1)$ which satisfies the condition given in Theorem 5.6. We observe in Fig. 5.12 that the failure can hardly spread out.

5.6 Applications in Failure Control

Our analysis for failure spread provides a mathematical framework to evaluate the resilience of large wireless networks against correlated failures. Given a network with node density λ, we may sample the nodes to estimate their failure impact radius distribution $f(x)$ and their failure propagation probability $g(x)$, and determine the characteristic regime (percolation or non-percolation) that λ, $f(x)$ and $g(x)$ fall into. This evaluation allows us to predict the potential impact on the network when failure occurs.

Besides network resilience evaluation, our analysis also indicates a few strategies to prevent correlated failures from wide spreading in the network. Known from the analysis, failure becomes unlikely to percolate if we reduce the impact radius r and the failure connection probability p when the node density λ is given. We are hence able to control the failure spread via bounding r and p. We next discuss the failure control strategies respectively for the one-time failures and the persistent failures.

For one-time failures, we contain the failure spread by limiting either r or p. To limit r, each node in the network is configured not to execute any suspicious command received from nodes located beyond a certain distance, which effectively reduces the impact radius of each failed node. To limit p, we need to sample and test some portion of nodes

in the network to ensure that these tested nodes are not vulnerable to the type of failure that we are concerned with, such that the failure probability of an arbitrary node in the network is controllably low after the test.

For persistent failures, however, we may not be able to reduce r or p separately. In the example of traffic overloading, if we require the routing logic to limit the path reparation radius to a small value, the failure probability of each node within this radius might be high as each node is expected to receive a large share of the re-routed traffic. In other words, r and p are coupled. Therefore, the best strategy to reduce the degree of failure correlations is to incorporate load balancing considerations into the routing protocol design.

5.7 Summary

In this chapter, we have studied the spread of correlated failures in large wireless networks. When there exist causal relations, a single failure may initiate a cascading sequence of related failures that severely impair the correct network functioning. In order to understand the occurrence of large-scale failure spreading, we model the contagiousness of various types of failures into two generic correlation functions and determine their characteristic regimes in which failures will and will not spread to the entire network. Our results provide a quantified answer on the relation between failure correlations and failure pervasiveness. They match the intuition that failures spread to a large area when they have strong correlations. By using a generalized model, our results are applicable to the understanding of a wide range of failure scenarios. The work in this chapter helps network designers and operators evaluate and improve the resilience of large wireless networks against correlated failures.

Chapter 6

Conclusion and Future Directions

In this dissertation, we have presented our research results regarding the performance and resilience of large-scale wireless networks in four fundamental aspects: the information propagation speed, the communication capacity, the topological stability, and the failure resilience. Next, we summarize our results and discuss the possible future extensions.

6.1 Conclusion

This dissertation has focused on the modeling, analysis and understanding of large-scale multi-hop wireless networks on their fundamental performance and resilience perspectives. We first investigated the minimum packet delay in large wireless networks via our new metric, the information propagation speed, in Chapter 2. We then studied the practical link scheduling strategies in large wireless networks that achieves the order optimal network capacity with significantly reduced protocol overhead in Chapter 3. On the network resilience aspects, we analyzed the impact of random node mobility on the architectural stability of hierarchical wireless networks in Chapter 4 and characterized the spread of correlated failures in Chapter 5. We next recapitulate and highlight our major findings.

In Chapter 2, we defined a new metric to study the minimum packet delay in large wireless networks, the information propagation speed, which is the geographical distance traveled through by a packet in a given amount of time. As communications in a network can be viewed as transporting a packet from one physical location to another, our defined speed metric effectively characterizes the packet delay in a large network. Our analysis identified three optimal conditions that maximize the information propagation speed re-

garding the distribution and selection of the relay nodes on the communication paths. We further considered the requirement for network connectivity and obtained results on the speed upper bounds in different noise and interference environments. Because nodes are randomly located in a large wireless network, we also determined the speed difference from its bound when the selected relay nodes are not at the ideal locations, which gives us further understanding of the fastest information propagation in realistic network settings. Finally, we proposed a new geographic routing algorithm that minimizes the packet transmission delays in wireless networks. Our work advances our understanding of the minimum packet delay in large wireless networks.

In Chapter 3, we explored the possibility of using localized link scheduling schemes to achieve order optimal capacity in large wireless networks. Due to wireless interference, nodes in a wireless network cannot transmit simultaneously if they are close to each other. Link scheduling is thus widely used in wireless networks to avoid transmission collisions. The existing studies on capacity maximization in wireless networks all have relied on global link scheduling schemes which coordinate and synchronize node transmissions across the entire network. Our analysis however shows that we can achieve comparable network capacity by dividing a large network into small autonomous regions and scheduling the transmissions in each region independently from others. Specifically, we obtained the sufficient and necessary conditions on our approach of network decomposition and localized scheduling for capacity maximization. By following the analytical results as a theoretical guideline, we also designed a distributed network partition protocol to generate the autonomous regions and a localized link scheduling scheme to coordinate transmissions in each region. Our design maximizes the network capacity with significantly less complexity and overhead than the global scheduling strategies.

In Chapter 4, we examined the network architectural stability problem in large-scale mobile wireless networks. When nodes are not static, the network topology experiences constant changes over time. The communications in the network suffer from the unstable topology as the traffic sessions are interrupted from time to time and the path establishment delays and overheads are increased. We chose a general mobility model to capture the typical node movements and analyzed the topological stability of the hierarchical network architecture under the impact of node mobility. To quantify the topological stability, we defined three metrics, namely, the cluster lifetime, the link lifetime and the path lifetime, and derived their respective upper bounds. We then designed a new node clustering algorithm

and a new hierarchical routing protocol to extend the network topological lifetime to the maximum, based on our discovery that the network topology is stabilized most when we eliminate all the unnecessary structural changes. Simulation showed that our proposed algorithm and protocol are effective in preserving network topologies.

In Chapter 5, we characterized the spread of correlated failures in large wireless networks. As failures happen inevitably in any communication systems and networks, it is imperative for us to gain an insightful understanding of the potential impact created by failures on the overall network performance. Our work specifically focused on the correlated failures which threaten the normal network functioning much more severely than the random and independent failures. We considered two categories of failures, the one-time failures in which each node is at risk of failure only once and the persistent failures in which each node may fail every time affected by a failed node. We first formulated the contagiousness of correlated failures as general functions and then utilized the percolation theory to determine the trend of failure spread in the long run. Our results established a quantitative relation between the strength of failure correlation and the occurrence of wide-spread failure propagations. We determined the characteristic regimes for the case that failure will spread out if each failure is able to infect sufficiently many other nodes and the case that failure will not spread out if the infection is limited. Our characterization of failure spread suggested strategies to contain the failure propagation and improve the network resilience.

6.2 Future Directions

The work presented in this dissertation is only part of the efforts in understanding and maximizing the performance of large-scale wireless networks. In order to construct effective, efficient, reliable, trustworthy and customizable wireless networks, we may extend our study in the following directions.

Our current work evaluates the performance and resilience of wireless networks from various stand-alone perspectives. A joint study on the interaction of different performance aspects will provide more comprehensive understanding of the large-scale wireless networks in general circumstances. For example, the joint optimization of packet delay and network capacity needs a further study. Moreover, with the uncertainty introduced by node mobility and failure, the best strategy for fastest information delivery and network capacity maximization may need a thorough investigation.

When a wireless network is designed for specific tasks, it may have particular features that affect its performance optimization strategies. For example, the mesh networks and the sensor networks differ from the general purpose wireless networks on their network structures and communication requirements. Besides, hybrid wireless networks that consist of different types of sub-networks and wireless networks inter-connected with wired networks such as Internet are also commonly used. Developing network-specific and task-specific performance maximization strategies will be another interesting research problem.

Due to the communication openness, wireless networks are threatened by various potential adversaries that may intercept, redirect, falsify, discard or jam the legitimate traffic in the network. In addition, sophisticated adversaries may even exploit the vulnerabilities in network protocols to disconnect the users or partition the network. Withstanding potential attacks and establishing service trustworthiness are a prerequisite for constructing usable wireless networks. The incorporation of information security and service availability mechanisms into the design and operation of wireless networks is therefore both beneficial and necessary.

Users in a network display strong movement and communication patterns that are closely related to their occupations, social contacts, living styles, personal preferences, and etc. For example, a typical user may commute between home and workplace everyday and a group of users may have similar working schedules or common interests in certain tasks. The specific user features may be exploited to optimize the design and operation of wireless networks and furthermore to customize the services provided to each user. Personalized and intelligent services will be highly desired in the current and future wireless networks.

Bibliography

[1] P. Gupta and P.R. Kumar, "The Capacity of Wireless Networks", *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, March 2000.

[2] M. Grossglauser and D. Tse, "Mobility Increases the Capacity of Ad Hoc Wireless Networks", *IEEE/ACM Transactions on Networking*, vol. 10, no. 4, pp. 477–486, August 2002.

[3] B. Liu, Z. Liu, and D. Towsley, "On the Capacity of Hybrid Wireless Networks", *Proc. of IEEE Infocom*, March 2003.

[4] A. Ozgur, O. Leveque, and D. Tse, "Hierarchical Cooperation Achieves Linear Capacity Scaling in Ad Hoc Networks", *Proc. of IEEE Infocom*, May 2007.

[5] H. Zhang and J.C. Hou, "Capacity of Wireless Ad-Hoc Networks under Ultra Wide Band with Power Constraint", *Proc. of IEEE Infocom*, March 2005.

[6] M. Kodialam and T. Nandagopal, "Characterizing Achievable Rates in Multi-Hop Wireless Mesh Networks with Orthogonal Channels", *IEEE/ACM Transactions on Networking*, vol. 13, no. 4, pp. 868–880, August 2005.

[7] R. Zheng, "Information Dissemination in Power-Constrained Wireless Networks", *Proc. of IEEE Infocom*, April 2006.

[8] M. Garetto, P. Giaccone, and E. Leonardi, "On the Capacity of Ad Hoc Wireless Networks Under General Node Mobility", *Proc. of IEEE Infocom*, May 2007.

[9] U.C. Kozat and L. Tassiulas, "Throughput Capacity of Random Ad Hoc Networks with Infrastructure Support", *Proc. of ACM MobiCom*, September 2003.

[10] M. Gastpar and M. Vetterli, "On the Capacity of Wireless Networks: the Relay Case", *Proc. of IEEE Infocom*, June 2002.

[11] R. Negi and A. Rajeswaran, "Capacity of Power Constrained Ad-Hoc Networks", *Proc. of IEEE Infocom*, March 2004.

[12] A.E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, "Throughput-Delay Trade-off in Wireless Networks", *Proc. of IEEE Infocom*, March 2004.

[13] G. Sharma, R. Mazumdar, and N. Shroff, "Delay and Capacity Trade-offs in Mobile Ad Hoc Networks: A Global Perspective", *Proc. of IEEE Infocom*, April 2006.

[14] M. Garetto, P. Giaccone, and E. Leonardi, "Capacity Scaling in Delay Tolerant Networks with Heterogeneous Mobile Nodes", *Proc. of ACM MobiHoc*, September 2007.

[15] W. Chen, M.J. Neely, and U. Mitra, "Energy Efficient Scheduling with Individual Packet Delay Constraints: Offline and Online Results", *Proc. of IEEE Infocom*, May 2007.

[16] R. Cohen and B. Kapchits, "An Optimal Algorithm for Minimizing Energy Consumption while Limiting Maximum Delay in a Mesh Sensor Network", *Proc. of IEEE Infocom*, May 2007.

[17] D. Rajan, "Power Efficient Delay Allocation in Multihop Wireless Networks", *IEEE Transactions on Vehicular Technology*, vol. 56, no. 4, pp. 1813–1825, July 2007.

[18] W. Lai and I.C. Paschalidis, "Sensor Network Minimal Energy Routing with Latency Guarantees", *Proc. of ACM MobiHoc*, September 2007.

[19] M.M. Krunz and J.G. Kim, "Fluid Analysis of Delay and Packet Discard Performance for QoS Support in Wireless Networks", *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 2, pp. 384–395, February 2001.

[20] S. Wang, R. Nathuji, R. Bettati, and W. Zhao, "Providing Statistical Delay Guarantees in Wireless Networks", *Proc. of IEEE International Conference on Distributed Computing Systems (ICDCS)*, March 2004.

[21] J. Liebeherr, D.E. Wrege, and D. Ferrari, "Exact Admission Control for Networks with a Bounded Delay Service", *IEEE/ACM Transactions on Networking*, vol. 4, no. 6, pp. 885–901, December 1996.

[22] M. Verloop and S. Borst, "Heavy-Traffic Delay Minimization in Bandwidth-Sharing Networks", *Proc. of IEEE Infocom*, May 2007.

[23] Y. Yang and R. Kravets, "Achieving Delay Guarantees in Ad Hoc Networks through Dynamic Contention Window Adaptation", *Proc. of IEEE Infocom*, April 2006.

[24] A. Bader and E. Ekici, "Throughput and Delay Optimization in Interference-Limited Multihop Networks", *Proc. of ACM MobiHoc*, May 2006.

[25] B. Liang and M. Dong, "Balancing Distance and Lifetime in Delay Constrained Ad Hoc Networks", *Proc. of ACM MobiHoc*, May 2006.

[26] P. Djukic and S. Valaee, "Link Scheduling for Minimum Delay in Spatial Re-use TDMA", *Proc. of IEEE Infocom*, May 2007.

[27] D. Chafekar, V.S.A. Kumar, M.V. Marathe, S. Parthasarathy, and A. Srinivasan, "Cross-Layer Latency Minimization in Wireless Networks with SINR Constraints", *Proc. of ACM MobiHoc*, September 2007.

[28] M. Penrose, *Random Geometric Graphs*, Oxford University Press, 2003.

[29] M. Schwartz, *Mobile Wireless Communications*, Cambridge University Press, 2005.

[30] C.E. Shannon, "A Mathematical Theory of Communications", *Bell Syst. Tech. J.*, pp. 379–423 (Part 1); pp. 623–656 (Part 2), July 1948.

[31] S. Lin and D.J. Costello, *Error Control Coding: Fundamentals and Applications*, 2nd ed., Pearson Prentice Hall, 2004.

[32] M. Penrose, "The Longest Edge of the Random Minimal Spanning Tree", *The Annals of Applied Probability*, vol. 7, no. 2, pp. 340–361, May 1997.

[33] B. Karp and H.T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks", *Proc. of ACM MobiCom*, August 2000.

[34] S.R. Kulkarni and P. Viswanath, "A Deterministic Approach to Throughput Scaling in Wireless Networks", *IEEE Transactions on Information Theory*, vol. 50, no. 6, pp. 1041–1049, June 2004.

[35] A. Keshavarz-Haddad, V. Ribeiro, and R. Riedi, "Broadcast Capacity in Multihop Wireless Networks", *Proc. of ACM MobiCom*, September 2006.

[36] A. Keshavarz-Haddad and R. Riedi, "Bounds for the Capacity of Wireless Multihop Networks Imposed by Topology and Demand", *Proc. of ACM MobiHoc*, September 2007.

[37] S. Shakkottai, X. Liu, and R. Srikant, "The Multicast Capacity of Large Multihop Wireless Networks", *Proc. of ACM MobiHoc*, September 2007.

[38] X.Y. Li, S.J. Tang, and O. Frieder, "Multicast Capacity for Large Scale Wireless Ad Hoc Networks", *Proc. of ACM MobiCom*, September 2007.

[39] Z. Wang, H.R. Sadjadpour, and J.J. Garcia-Luna-Aceves, "A Unifying Perspective on the Capacity of Wireless Ad Hoc Networks", *Proc. of IEEE INFOCOM*, April 2008.

[40] M. Franceschetti, O. Dousse, D.N.C. Tse, and P. Thiran, "Closing the Gap in the Capacity of Wireless Networks via Percolation Theory", *IEEE Transactions on Information Theory*, vol. 53, no. 3, pp. 1009–1018, March 2007.

[41] S. Li, Y. Liu, X.Y. Li, "Capacity of Large Scale Wireless Networks Under Gaussian Channel Model", *Proc. of ACM MobiCom*, September 2008.

[42] G. Brar, D.M. Blough, and P. Santi, "Computationally Efficient Scheduling with the Physical Interference Model for Throughput Improvement in Wireless Mesh Networks", *Proc. of ACM MobiCom*, September 2006.

[43] R. Mahjourian, F. Chen, R. Tiwari, M. Thai, H. Zhai, and Y. Fang, "An Approximation Algorithm for Conflict-Aware Broadcast Scheduling in Wireless Ad Hoc Networks", *Proc. of ACM MobiHoc*, May 2008.

[44] G. Sharma, R.R. Mazumdar, and N.B. Shroff, "On the Complexity of Scheduling in Wireless Networks", *Proc. of ACM MobiCom*, September 2006.

[45] T. Moscibroda, R. Wattenhofer, and A. Zollinger, "Topology Control Meets SINR: The Scheduling Complexity of Arbitrary Topologies", *Proc. of ACM MobiHoc*, May 2006.

[46] O. Goussevskaia, Y.A. Oswald, and R. Wattenhofer, "Complexity in Geometric SINR", *Proc. of ACM MobiHoc*, September 2007.

[47] W. Wang, Y. Wang, X.Y. Li, W.Z. Song, and O. Frieder, "Efficient Interference-Aware TDMA Link Scheduling for Static Wireless Networks", *Proc. of ACM MobiCom*, September 2006.

[48] E. Modiano, D. Shah, and G. Zussman, "Maximizing Throughput in Wireless Networks via Gossiping", *Proc. of ACM SIGMetrics/Performance*, June 2006.

[49] P. Chaporkar, K. Kar, and S. Sarkar, "Throughput Guarantees Through Maximal Scheduling in Wireless Networks", *Proc. of 43rd Allerton Conf. on Commun., Control, and Comp.*, September 2005.

[50] L. Chen, S.H. Low, M. Chiang, and J.C. Doyle, "Cross-Layer Congestion Control, Routing and Scheduling Design in Ad Hoc Wireless Networks", *Proc. of IEEE INFOCOM*, April 2006.

[51] X. Lin and N.B. Shroff, "The Impact of Imperfect Scheduling on Cross-Layer Rate Control in Wireless Networks", *Proc. of IEEE INFOCOM*, March 2005.

[52] C. Joo, X. Lin, and N.B. Shroff, "Understanding the Capacity Region of the Greedy Maximal Scheduling Algorithm in Multi-Hop Wireless Networks", *Proc. of IEEE INFOCOM*, April 2008.

[53] C. Joo and N.B. Shroff, "Performance of Random Access Scheduling Schemes in Multi-Hop Wireless Networks", *Proc. of IEEE INFOCOM*, May 2007.

[54] D. Chafekar, D. Levin, V.S.A. Kumar, M.V. Marathe, S. Partha-sarathy, and A. Srinivasan, "Capacity of Asynchronous Random-Access Scheduling in Wireless Networks", *Proc. of IEEE INFOCOM*, April 2008.

[55] O. Dousse and P. Thiran, "Connectivity vs Capacity in Dense Ad Hoc Networks", *Proc. of IEEE INFOCOM*, March 2004.

[56] A. Okabe, B. Boots, K. Sugihara, and S.N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd ed., Wiley Series in Probability and Statistics, 2000.

[57] A. Ephremides, J.E. Wieselthier, and D.J. Baker, "A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling," *Proc. of IEEE*, vol. 75, pages 56–73, 1987.

[58] M. Gerla, and J.T-C. Tsai, "Multicluster, Mobile, Multimedia Radio Network," *Wireless Networks*, vol. 1, pages 255–265, October 1995.

[59] S. Basagni, "Distributed Clustering for Ad Hoc Networks," *Proc. of ISPAN, International Symposium on Parallel Architectures, Algorithms, and Networks*, pages 310–315, June 1999.

[60] V. Kawadia, and P.R. Kumar, "Power Control and Clustering in Ad Hoc Networks," *Proc. of IEEE Infocom*, pages 459–469, 2003.

[61] R. Ramanathan, and M. Steenstrup, "Hierarchically-Organized Multihop Mobile Wireless Networks for Quality-of-Service Support," *Mobile Networks and Applications*, pages 101–119, 1998.

[62] X. Hong, M. Gerla, Y. Yi, K. Xu, and T.J. Kwon, "Scalable Ad Hoc Routing in Large, Dense Wireless Networks Using Clustering and Landmarks," *Proc. of IEEE ICC*, pages 3179–3185, 2002.

[63] O. Younis and S. Fahmy, "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks," *IEEE Transactions on Mobile Computing*, 3(4):366–379, October-December 2004.

[64] C.F. Hsin and M. Liu, "Partial Clustering: Maintaining Connectivity in a Low Duty-Cycled Dense Wireless Sensor Network," *Proc. of IPDPS, the IEEE International Parallel and Distributed Processing Symposium*, pages 1–8, 2005.

[65] L. Ramachandran, M. Kapoor, A. Sarkar, and A. Aggarwal, "Clustering Algorithms for Wireless Ad Hoc Networks," *Proc. of ACM DIALM Workshop*, pages 54–63, 2000.

[66] T.C. Hou and T.J. Tsai, "An Access-Based Clustering Protocol for Multihop Wireless Ad Hoc Networks," *IEEE Journal on Selected Areas in Communications*, 19(7):1201–1210, July 2001.

[67] B. An, and S. Papavassiliou, "A Mobility-Based Clustering Approach to Support Mobility Management and Multicast Routing in Mobile Ad-Hoc Wireless Networks," *International Journal of Network Management*, pages 387–395, 2001.

[68] P. Basu, N. Khan, and T.D.C. Little, "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks," *Proc. of IEEE ICDCSW*, pages 413–418, April 2001.

[69] A.B. McDonald, and T.F. Znati, "A Mobility-Based Framework for Adaptive Clustering in Wireless Ad Hoc Networks," *IEEE Journal on Selected Areas in Communications*, 17(8):1466–1487, August 1999.

[70] M. Chatterjee, S.K. Das, and D. Turgut, "WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks," *Cluster Computing*, 5, 193– 204, 2002.

[71] L. Bao, and J.J. Garcia-Luna-Aceves, "Topology Management in Ad Hoc Networks," *Proc. of ACM MobiHoc*, pages 129–140, June 2003.

[72] S. Sivavakeesar, and G. Pavlou, "Scalable Location Services for Hierarchically Organized Mobile Ad Hoc Networks," *Proc. of ACM MobiHoc*, pages 217–228, 2005.

[73] C.-C. Chiang, H.-K. Wu, W. Liu, and M. Gerla, "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel," *Proc. of IEEE SICON*, pages 197–211, April 1997.

[74] R. Ghosh, and S. Basagni, "Limiting the Impact of Mobility on Ad Hoc Clustering," *Proc. of ACM PE-WASUN*, pages 197–204, October 2005.

[75] G. Pei, M. Gerla, and X. Hong, "LANMAR: Landmark Routing for Large Scale Wireless Ad Hoc Networks with Group Mobility," *Proc. of ACM MobiHoc*, pages 11–18, 2000.

[76] N. Zhou, and A.A. Abouzeid, "Routing in Ad Hoc Networks: A Theoretical Framework with Practical Implications," *Proc. of IEEE Infocom*, pages 1240–1251, 2005.

[77] M. Jiang, J. Li, and Y.C. Tay, "Cluster Based Routing Protocol (CBRP)," IETF MANET Working Group, Internet Draft, July 1999.

[78] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: A Core- Extraction Distributed Ad Hoc Routing Algorithm," *IEEE Journal on Selected Areas in Communications*, 17(8):1454–1465, August 1999.

[79] E. Belding-Royer, "Hierarchical Routing in Ad Hoc Mobile Networks," *Wireless Communications and Mobile Computing*, vol. 2, issue 5, pages 515–532, 2002.

[80] K. Xu, X. Hong, and M. Gerla, "Landmark Routing in Ad Hoc Networks with Mobile Backbones," *Journal of Parallel and Distributed Computing*, 63(2):110–122, February 2003.

[81] P. Krishna, N.H. Vaidya, M. Chatterjee, D.K. Pradhan, "A Cluster-based Approach for Routing in Dynamic Networks," *ACM SIGCOMM Computer Communication Review*, 1997.

[82] Y. Chen, A.L. Liestman, "Approximating Minimum Size Weakly-Connected Dominating Sets for Clustering Mobile Ad Hoc Networks," *Proc. of ACM MobiHoc*, pages 165–172, June 2002.

[83] M.R. Pearlman and Z.J. Haas, "Determining the Optimal Configuration for the Zone Routing Protocol," *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 8, pages 1395–1414, August 1999.

[84] M. Gerharz, C. Waal, M. Frank, and P. Martini, "Link Stability in Mobile Wireless Ad Hoc Networks," *Proc. of IEEE LCN*, 2002.

[85] N. Sadagopan, F. Bai, B. Krishnamachari, and A. Helmy, "PATHS: Analysis of PATH Duration Statistics and their Impact on Reactive MANET Routing Protocols," *Proc. of ACM MobiHoc*, June 2003.

[86] P. Samar, and S.B. Wicker, "On the Behavior of Communication Links of a Node in a Multi-Hop Mobile Environment," *Proc. of ACM MobiHoc*, May 2004.

[87] V. Lenders, J. Wagner, and M. May, "Analyzing the Impact of Mobility in Ad Hoc Networks," *Proc. of ACM REALMAN*, May 2006.

[88] E. Royer, P.M. Melliar-Smith, and L. Moser, "An Analysis of the Optimum Node Density for Ad Hoc Mobile Networks," *Proc. of IEEE ICC*, 2001.

[89] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research," *Wireless Communications and Mobile Computing*, vol. 2, issue 5, pages 483–502, 2002.

[90] D.B. Johnson and D.A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," *Mobile Computing*, 1996.

[91] P. Nain, D. Towsley, B. Liu, and Z. Liu, "Properties of Random Direction Models," in *Proc. of IEEE INFOCOM*, March 2005.

[92] C. Bettstetter, "Smooth is Better than Sharp: A Random Mobility Model for Simulation of Wireless Networks," in *Proc. of ACM International Workshop on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWiM)*, July 2001.

[93] B. Liang and Z. Haas, "Predictive Distance-Based Mobility Management for PCS Networks," in *Proc. of IEEE INFOCOM*, April 1999.

[94] J. Tian, J. Hahner, C. Becker, I. Stepanov, and K. Rotherme, "Graph-based Mobility Model for Mobile Ad Hoc Network Simulation," in *Proc. of the 35th Annual Simulation Symposium*, in cooperation with the IEEE Computer Society and ACM, 2002.

[95] F. Bai, N. Sadagopan, B. Krishnamachari, and A. Helmy, "Important: A Framework to Systematically Analyze the Impact of Mobility on Performance of Routing Protocols for Ad Hoc Networks," in *Proc. of IEEE INFOCOM*, 2003.

[96] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, "Scenario-Based Performance Analysis of Routing Protocols for Mobile Ad-Hoc Networks," in *Proc. of Mobicom*, pp. 195–206, 1999.

[97] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, "A Group Mobility Model for Ad Hoc Wireless Networks," *Proc. of ACM/IEEE MSWiM*, August 1999.

[98] The Network Simulator NS-2. http://www.isi.edu/nsnam/ns/.

[99] Z. Yu, Y. Wei, B. Ramkumar, and Y. Guan, "An Efficient Signature-Based Scheme for Securing Network Coding against Pollution Attacks", *Proc. of IEEE INFOCOM*, 2008.

[100] S. Stefanakos, "Reliable Routings in Networks with Generalized Link Failure Events", *IEEE/ACM Transactions on Networking*, vol. 16, no. 6, pp. 1331–1339, December 2008.

[101] B. Awerbuch, D. Holmer, C. Nita-Rotaru, and H. Rubens, "An On-Demand Secure Routing Protocol Resilient to Byzantine Failures", *Proc. of ACM WiSe*, 2002.

[102] T. Bu, A. Chen, S. Wiel, and T. Woo, "Design and Evaluation of a Fast and Robust Worm Detection Algorithm", *Proc. of IEEE INFOCOM*, 2006.

[103] V. Bhandari and N.H. Vaidya, "Reliable Broadcast in Wireless Networks with Probabilistic Failures", *Proc. of IEEE INFOCOM*, 2007.

[104] Y. Kim, W.C. Lau, M.C. Chuah, and H.J. Chao, "Packetscore: Statistics-Based Overload Control against Distributed Denial-of-Service Attacks", *Proc. of IEEE INFOCOM*, 2004.

[105] S. Yu and Y. Zhang, "R-Sentry: Providing Continuous Sensor Services against Random Node Failures", *Proc. of IEEE DSN*, 2007.

[106] S. Nelakuditi, S. Lee, Y. Yu, Z.L. Zhang, and C.N. Chuah, "Fast Local Rerouting for Handling Transient Link Failures", *IEEE/ACM Transactions on Networking*, vol. 15, no. 2, pp. 359–372, April 2007.

[107] R.S. Bhatia, M. Kodialam, T.V. Lakshman, and S. Sengupta, "Bandwidth Guaranteed Routing with Fast Restoration against Link and Node Failures", *IEEE/ACM Transactions on Networking*, vol. 16, no. 6, pp. 1321–1330, December 2008.

[108] P.R. Jelenkovic and J. Tan, "Dynamic Packet Fragmentation for Wireless Channels with Failures", *Proc. of ACM MobiHoc*, 2008.

[109] S. Jain, M. Demmer, R. Patra, and K. Fall, "Using Redundancy to Cope with Failures in a Delay Tolerant Network", *Proc. of ACM SIGCOMM*, 2005.

[110] F. Xing and W. Wang, "On the Critical Phase Transition Time of Wireless Multi-hop Networks with Random Failures", *Proc. of ACM MobiCom*, 2008.

[111] Z. Kong and E.M. Yeh, "Distributed Energy Management Algorithm for Large-Scale Wireless Sensor Networks", *Proc. of ACM MobiHoc*, 2007.

[112] R. Meester and R. Roy, *Continuum Percolation*, Cambridge University Press, 1996.

[113] G. Grimmett, *Percolation*, 2nd ed., Springer, 1999.

[114] M. Penrose, *Random Geometric Graphs*, Oxford University Press, 2003.

[115] Z. Kong and E.M. Yeh, "Connectivity and Latency in Large-Scale Wireless Networks with Unreliable Links", *Proc. of IEEE INFOCOM*, 2008.

[116] Z. Kong and E.M. Yeh, "On the Latency for Information Dissemination in Mobile Wireless Networks", *Proc. of ACM MobiHoc*, 2008.

[117] O. Dousse, P. Mannersalo, and P. Thiran, "Latency of Wireless Sensor Networks with Uncoordinated Power Saving Mechanisms", *Proc. of ACM MobiHoc*, 2004.