

**MODELING AND ANALYSIS OF THE PERFORMANCE
OF COLLABORATIVE WIRELESS AD-HOC
NETWORKS: AN INFORMATION-THEORETIC
PERSPECTIVE**

A Thesis
Presented to
The Academic Faculty

by

Ramanan Subramanian

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
December 2009

MODELING AND ANALYSIS OF THE PERFORMANCE OF COLLABORATIVE WIRELESS AD-HOC NETWORKS: AN INFORMATION-THEORETIC PERSPECTIVE

Approved by:

Professor Faramarz Fekri, Advisor
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Professor Raghupathy Sivakumar
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Professor Gordon Stuber
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Professor Mostafa Ammar
College of Computing
Georgia Institute of Technology

Professor Ed Coyle
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Date Approved: 23 October 2009

To my loving family and friends
whose good wishes made this possible

ACKNOWLEDGEMENTS

Having successfully defended my PhD thesis and finding myself reflecting about the past six years of my time in Atlanta, I plead guilty to not being able to find enough words to every one who has motivated me and touched my life during those years. Without the blessing of their good association, it would have been really difficult for someone coming from a conservative suburban region in India to start a fresh life as a graduate student in urban America. Though there are many whom I have to thank, I will first start with my thesis advisor Dr. Faramarz Fekri. Without his assistance and support, it would have been impossible for me to have come this far, both career-wise and personally. I also thank him for the facilities he has provided and for giving me an opportunity to explore and further my knowledge in the field. I would next like to thank the distinguished faculty members who agreed to serve in my proposal and dissertation committees, who, by name, are: Dr. Raghupathy Sivakumar, Dr. Gordon Stuber, Dr. Xiaoli Ma, Dr. Mostafa Ammar, and Dr. Edward Coyle. I cannot thank them enough for gracing my final examinations with their presence and for their valuable feedback and timely advice. I would then like to thank my fellow labmate Badri Vellambi, who likewise graduated with a PhD from Georgia Tech last fall. The collaboration and guidance in my research that I received from Badri is unmatched.

I would then like to mention about the great time I had from living with roommates as fellow graduate students. Had they not agreed to share a home, it would have been impossible for me to manage the pressures at work along with the household chores, not to mention the financial advantage arising from that. Hence, I would like to thank them by name: Arun Rambhatla, Roshan Menon, and Tapobrata Bandyopadhyay,

and of course, my younger brother Arunkumar Subramanian who is pursuing his graduate education also in the School of ECE, and last but not the least, Badri Vellambi.

I am out of words to express my gratitude towards the friends belonging to the Greater Atlanta Indian community. The foremost of them are Badri who was also my fellow labmate. I cannot forget the many long sessions we used to spend at home and work discussing Indian Classical Music and general philosophy, apart from research-related discussions which were very valuable. I cannot thank enough my friends Subhashini, her husband Shiva, and their one-and-half-year-old daughter Srividya for their great company. I cannot count the awesome weekends of fun that we had in their place in Norcross. Also many many thanks to Venkat Ramachandran for introducing Subhashini and Shiva to me and Badri, in addition to numerous rides to various places in Atlanta when we needed them. Also, I cannot afford to skip mentioning Ramesh and his great company during the 3:00 pm coffee sessions with Badri and me.

I also thank my friends Sandeep Kakumanu, Nischint Rajmohan, Yash Kochar, Saptharishi and his wife Shubha, who were fellow graduate students during my stay in Atlanta and with whom I got to participate in those fun-filled “poker” and “monopoly” sessions during the weekends, and for the numerous trips and outings that we did together. The support, encouragement that they gave and the fun that they shared with me during weekends was of immense help in overcoming the stressful times during my graduate education.

Last but not the least, I thank the Almighty, who is compassion personified, the eternal abode of wisdom, strength, sovereignty, vigour, energy, and splendor, the One Mother, Father, Preceptor, and Friend to all, for showering his unlimited grace on me during the last six years and for continuing to do so. I only pray to him to bless one and all with the comforts and happiness I have been fortunate enough to receive.

TABLE OF CONTENTS

| | |
|--|-----|
| DEDICATION | iii |
| ACKNOWLEDGEMENTS | iv |
| LIST OF TABLES | xi |
| LIST OF FIGURES | xii |
| SUMMARY | xv |
| I INTRODUCTION AND RELATED WORK | 1 |
| 1.1 Fundamental Performance Limits in Distributed Wireless Sensor Networks | 4 |
| 1.1.1 Lifetime Optimization in Wireless Sensor Networks | 5 |
| 1.1.2 Latency Analysis for Wireless Sensor Networks | 7 |
| 1.2 Performance Characterization of Sparse Mobile Ad-Hoc Networks | 10 |
| 1.2.1 Markovian Approach for Performance Analysis of SPMANETs | 12 |

VOLUME I

| | |
|---|----|
| II LIFETIME MAXIMIZATION IN DISTRIBUTED WIRELESS SENSOR NETWORKS | 14 |
| 2.1 Assumptions and Notations | 15 |
| 2.2 Preliminary Analysis | 18 |
| 2.3 Proposed solution | 21 |
| 2.3.1 Uniform Distribution | 21 |
| 2.3.2 Optimum Non-Uniform Distribution | 22 |
| 2.4 Simulation Results | 25 |
| 2.5 Significance of our Results | 27 |
| 2.6 Summary | 28 |
| III LATENCY ANALYSIS FOR CONVERGECAST COMMUNICATION IN A DISTRIBUTED SENSOR NETWORK | 29 |
| 3.1 Analysis of Network Latency for Grid Sensor Networks | 31 |

| | | |
|-------|--|----|
| 3.2 | Simulation Results | 37 |
| 3.3 | Extension to Uniform Random Deployment | 39 |
| 3.3.1 | Path Spectrum Estimation | 42 |
| 3.3.2 | Delay Computation for a Given Path Length | 43 |
| 3.4 | Simulation Results for Uniform Random Deployment | 45 |
| 3.5 | Throughput Capacity of Sensor Networks | 46 |
| 3.6 | Summary | 48 |

VOLUME II

| | | |
|-------|---|----|
| IV | PRELIMINARY APPROACHES FOR MARKOV-CHAIN ANALYSIS OF SPMANETS | 49 |
| 4.1 | Motivation | 53 |
| 4.1.1 | Finite-Buffer Considerations in SPMANETs | 53 |
| 4.2 | Limitations of Poisson-process Modeling | 54 |
| 4.3 | Models and Definitions | 56 |
| 4.3.1 | Network and node mobility model | 56 |
| 4.3.2 | Packet transfer protocols | 58 |
| 4.3.3 | Contention resolution | 58 |
| 4.4 | Description of the General Approach | 59 |
| 4.5 | Two-hop single-unicast with immobile source and destination | 61 |
| 4.6 | Two-hop single-unicast with mobile source and destination | 68 |
| 4.7 | Simulation Results | 72 |
| 4.8 | Summary | 76 |
| V | EVOLVING A GENERALIZED APPROACH FOR THE ANALYSIS OF SPMANETS | 78 |
| 5.1 | Network Model | 79 |
| 5.1.1 | Definitions | 79 |
| 5.1.2 | Interference Model and Contention Resolution | 80 |
| 5.1.3 | Mobility Model | 81 |
| 5.1.4 | Two-Hop Single-Copy Routing | 81 |

| | | |
|-------|---|-----|
| 5.2 | Details of the Analysis | 82 |
| 5.2.1 | Notations | 82 |
| 5.2.2 | Overview of the Concept | 82 |
| 5.2.3 | Embedded Markov Chain and Chain-collapsing | 83 |
| 5.3 | Analysis of Single Source-Destination Unicast | 86 |
| 5.3.1 | Contention Analysis and Derivation of Mobility Parameters | 87 |
| 5.4 | Analysis and Simulation Results | 93 |
| 5.4.1 | Case 1: Random Walk Model | 93 |
| 5.4.2 | Case 2: Random Waypoint Model | 98 |
| 5.4.3 | Case 3: Restricted Random Waypoint Model | 99 |
| 5.5 | Summary | 100 |
| VI | ANALYSIS OF MULTIPLE-UNICAST STRATEGIES | 102 |
| 6.1 | Notations and Assumptions | 103 |
| 6.1.1 | Contention Resolution | 104 |
| 6.1.2 | State-space Description and Throughput | 104 |
| 6.2 | Setting up the embedded chain | 105 |
| 6.2.1 | Construction of Embedded Chain | 105 |
| 6.2.2 | Enumeration of Subsets | 106 |
| 6.2.3 | Steady-state Analysis | 107 |
| 6.2.4 | Steady-State Distribution for the Collapsed Chain $\Gamma_{\tau, v'}$ | 111 |
| 6.2.5 | Discussion of Analytical Results | 115 |
| 6.3 | Simulation Results | 117 |
| 6.3.1 | Random-Walk Mobility | 117 |
| 6.3.2 | Restricted-Random-Waypoint Mobility | 118 |
| 6.4 | Summary | 120 |
| VII | PERFORMANCE OF MULTIHOP ROUTING WITH BACK-PRESSURE POLICY | 121 |
| 7.1 | Network Model | 122 |
| 7.1.1 | Back-pressure-based Multihop Routing | 122 |

| | | |
|------------|---|-----|
| 7.1.2 | Contention Resolution | 123 |
| 7.2 | Analysis of Multihop Routing | 123 |
| 7.2.1 | Computation of Transition Probabilities and Steady-State Distributions for the Γ_v -Chain | 125 |
| 7.2.2 | Contention Analysis and Mobility Parameters | 128 |
| 7.3 | Simulation Results and Discussion | 131 |
| 7.3.1 | Discussion of Results | 131 |
| 7.4 | Summary | 134 |
| VIII | PERFORMANCE OF NETWORK-CODED MULTICAST | 136 |
| 8.1 | Network Model | 137 |
| 8.2 | Multicast Routing Protocols | 138 |
| 8.2.1 | Network-Coded Multicast | 138 |
| 8.2.2 | Simple Custodial Multicast | 139 |
| 8.3 | Analysis of Network-Coded Multicast | 140 |
| 8.3.1 | The “Stream Separation” Argument | 140 |
| 8.3.2 | Preliminary Steps in Analysis | 141 |
| 8.3.3 | Construction of the Occupancy Matrix and the Complemen- tary Occupancy Vector | 142 |
| 8.3.4 | Construction of the Network-level Chain | 145 |
| 8.3.5 | Performance of Multicast with Blind Back-Pressure Policy | 148 |
| 8.4 | Simulation Results | 150 |
| 8.5 | Summary | 151 |
| IX | CONCLUSION OF THE THESIS | 153 |
| APPENDIX A | DERIVATION OF THE MEAN AND VARIANCE OF THE HOPWISE EUCLIDEAN ADVANCE (SECTION 3.3) | 157 |
| APPENDIX B | PROOF OF LEMMA 5 | 160 |
| APPENDIX C | PROOF OF LEMMA 11 | 161 |
| REFERENCES | | 164 |

| | |
|----------------|-----|
| VITA | 172 |
|----------------|-----|

LIST OF TABLES

| | | |
|---|---|-----|
| 1 | Simulation Results | 25 |
| 2 | Storage capacities of some mobile devices | 53 |
| 3 | Enumeration of subsets for $m = 2$ | 107 |

LIST OF FIGURES

| | | |
|----|---|----|
| 1 | Distributed sensor network controlled by base station | 16 |
| 2 | State diagram for asynchronous sleep scheduling | 17 |
| 3 | Calculation of node traffic as a function of distance | 17 |
| 4 | Simulated node traffic behavior as a function of distance | 19 |
| 5 | Interference model | 19 |
| 6 | Network with uniform deployment of sensors | 24 |
| 7 | Network with optimal deployment of sensors | 24 |
| 8 | Plot of Node Power Expenditure (1000 nodes) (Uniform Deployment) | 26 |
| 9 | Plot of Node Power Expenditure (1000 nodes)(Optimum Non-Uniform Deployment) | 26 |
| 10 | Interference region in the grid network | 33 |
| 11 | Probability Density Function of inter arrival times for 8 bps source traffic and a different node location | 34 |
| 12 | Plot of node traffic vs hop distance for $m = 25$ | 35 |
| 13 | Latency (Time Slots $\times W$) as a function of the duty cycle | 37 |
| 14 | Plot of latency vs event frequency for different bandwidths | 38 |
| 15 | Idle Time Energy-Latency product as a function of duty cycle | 38 |
| 16 | Simulation : Latency vs Input Data Rate | 39 |
| 17 | Simulation : Latency vs Duty Cycle | 40 |
| 18 | Illustration of Geographical Routing | 43 |
| 19 | Comparison of Simulated and Estimated Path Spectrum | 44 |
| 20 | Simulation Results for Uniform Random Deployment - 1 | 46 |
| 21 | Simulation Results for Uniform Random Deployment - 2 | 47 |
| 22 | Comparison of Poisson method and exact analysis | 54 |
| 23 | Mobility Model | 57 |
| 24 | Markov chain for the two-hop grid network (ISD_2) | 62 |
| 25 | Solution of the ISD_2 chain by Potential method | 66 |

| | | |
|----|--|-----|
| 26 | State-space modification to include contention | 67 |
| 27 | Illustration of the Steady-State Distribution Solution obtained for the two-hop mobile source mobile destination unicast problem. | 71 |
| 28 | Throughput vs buffer size (fixed source/destination) | 73 |
| 29 | Throughput vs grid size (fixed source/destination) | 74 |
| 30 | Throughput vs Node population (fixed source/destination) | 74 |
| 31 | Throughput vs buffer size (mobile source/destination) | 75 |
| 32 | Throughput vs grid size (mobile source/destination) | 75 |
| 33 | Throughput vs Node population (mobile source/destination) | 76 |
| 34 | Collapsing a Markov Chain | 85 |
| 35 | Collapsed Markov chain for two-hop single unicast | 87 |
| 36 | Random-Walk mobility model for $R = 1$. Note: solid bubble = sending; concentric bubble = receiving; hollow bubble = idle | 93 |
| 37 | Throughput vs buffer size (Random Walk) | 96 |
| 38 | Throughput vs grid size (Random Walk) | 97 |
| 39 | Steady-state buffer distribution (Random Walk) | 97 |
| 40 | Throughput vs buffer size for Random Waypoint Mobility | 99 |
| 41 | Simulation results for the restricted random-waypoint mobility model | 100 |
| 42 | Network model for multiple-unicast analysis | 103 |
| 43 | Embedded Markov chain approach for contention analysis | 109 |
| 44 | Variation of $\alpha(m)$ with m | 116 |
| 45 | Simulation results for fixed- and floating-buffer multiple unicast un- der random walk mobility. Key: FBS = Fixed-buffer-segment; FB = Floating buffer | 118 |
| 46 | Multiple unicast simulation for restricted-random-waypoint mobility . | 119 |
| 47 | Composition of the Γ_v -chain illustrated for $B = 4$ | 125 |
| 48 | Simulation Results for the Random-Walk-on-Grid Mobility | 132 |
| 49 | Steady-state buffer-occupancy distribution for $B = 8$ | 133 |
| 50 | Steady-state buffer occupancy during link w/ destination node, for $n = 20$ and $n = 40$ | 133 |

| | | |
|----|--|-----|
| 51 | Simulation Results for the Restricted-Random-Waypoint Mobility . . | 134 |
| 52 | Network-Coded Multicast: Interaction of a relay node with the source and a destination node | 138 |
| 53 | Network-Coded Multicast: Interaction between two relay nodes . . . | 139 |
| 54 | General embedded chain for network-coded multicast. | 150 |
| 55 | Simulation results for varying buffer sizes | 151 |
| 56 | Simulation results for varying number of destinations | 152 |
| 57 | Computation of Next-Hop Advance PDF | 157 |

SUMMARY

This work focuses on the performance characterization of distributed collaborative ad-hoc networks, focusing on such metrics as the lifetime, latency, and throughput capacity of two such classes of networks. The first part concerns modeling and optimization of static Wireless Sensor Networks, specifically dealing with the issues of energy efficiency, lifetime, and latency. We analyze and characterize these performance measures and discuss various fundamental design tradeoffs. For example, energy efficiency in wireless sensor networks can only be improved at the cost of the latency (the delay incurred during communication). It has been clearly shown that improvement in energy efficiency through data aggregation increases the latency in the network. In addition, sleep-active duty cycling of nodes (devices constituting the network), a commonly employed mechanism to conserve battery lifetime in such networks, has adverse effects on their functionality and capacity. Hence these issues deserve a detailed study.

The second part of this work concerns performance modeling of Delay Tolerant Networks (DTNs) and Sparse Mobile Ad-Hoc Networks (SPMANETs) in general. We first investigate the effect of modern coding, such as the application of packet-level rateless codes, on the latency, reliability, and energy efficiency of the network. These codes provide us the means to break large messages into smaller packets thereby enabling efficient communication. The work then focuses on developing and formalizing an information-theoretic framework for Delay Tolerant- and other Sparse Mobile Networks. This is enabled by the use of an embedded-Markov-chain approach used

for complex queuing-theoretic problems. An important goal of this work is to incorporate a wide range of mobility models into the analysis framework. Yet another important question will be the effect of changing the mobility on the comparative performance of networking protocols. Lastly, the framework will be extended to various communication paradigms such as two-hop vs multi-hop routing, unicast, and multicast.

CHAPTER I

INTRODUCTION AND RELATED WORK

Breakthroughs in device technology, emphasis on pervasive computing, and the emergence of advanced communication protocols have made possible the exploration of several new networking paradigms for providing communication services. Often in these new paradigms, devices are deployed to accomplish a given communication task collaboratively. For example, they may route data through multiple hops, jointly aggregate and encode data, etc. Such networks – often deployed with very little infrastructure, communicate through the wireless medium and typically have very few access points, if any, to a backbone network such as the internet. Moreover, in applications such as defense, wildlife tracking, and environment monitoring, devices are deployed in hostile environments. In addition, device constraints such as low battery power, small memory sizes, small computation power, and mobility make communication in such networks quite challenging.

The emergence of new networking architectures described above has attracted a great deal of interest in the research community recently, in trying to understand their characteristics and thereby devise ways to optimize their performance. Though simulation and testbed evaluations are viable approaches today, such efforts often lead to very little insight into the tradeoffs between parameters and how they interplay to affect performance. Naturally, mathematical analysis of wireless networks has received much attention both in engineering and mathematical sciences research. In particular, analytical models for connectivity, coverage, capacity, and latency in wireless ad-hoc and sensor networks have been explored [1–5]. These studies have benefited from different areas such as probability theory, random graph theory, percolation theory,

geometry, algorithms, and complexity theory.

The first seminal work on the information-theoretic properties of wireless networks was conducted by Gupta *et. al.* in [1]. Here, the authors show that in a wireless network of n nodes within a given deployment region where arbitrary pairs of nodes communicate, the achievable throughput per node for randomly chosen destinations is given by $\Theta\left(\frac{W}{\sqrt{n \log n}}\right)$, where W is the bandwidth in bits per second. Moreover, even if nodes are optimally placed in a unit disc and the transmission powers are optimally chosen, the achievable per-node throughput is still only of the order $\Theta\left(\frac{W}{\sqrt{n}}\right)$. They have further shown that this fundamental constraint in capacity is due to the need for every node all over the domain to share the wireless channel with nodes in their neighborhood. In this path-breaking work, the authors have considered different models such as the protocol model and the Physical model for channel interference, as well as networks of arbitrary topology and uniformly-random deployments. They also showed that splitting the channel into several sub-channels also does not work. An important implication of this work was that it is better to build networks with smaller number of nodes.

While static networks were the primary concern of Gupta, Kumar *et. al.* in [1], further efforts on these lines were undertaken by Grossglauser, Tse, *et. al.* to study the same problem in a mobile scenario [6]. The primary finding of this research is that mobility can be exploited judiciously to increase the capacity of the network such that an overall per-node throughput of $O(1)$ is achievable. The authors note that the decrease in the per-node throughput otherwise is due to the fact that the source and destination nodes are quite far apart most of the time. Proposing to spread the traffic between intermediate nodes acting as relays gives diversity due to an increase in the number of available paths. However, this work did not consider the problem of delay. Though not relevant to real-time applications such as voice and video, the authors, for the first time in the literature, suggested that such a relaying approach could

greatly benefit delay-tolerant applications.

Recent work by El Gamal, Mammen, and Shah in [5,7] has elaborated on the above throughput-delay scaling. The authors considered a network the mobility of whose constituent nodes is described by a random-walk process on a toroidal grid. They showed that the optimal throughput-delay tradeoff is given by $D(n) = \Theta(nT(n))$ where $T(n)$ and $D(n)$ are the throughput and delay scalings, respectively. Moreover, the Grossglauser-Tse result of $O(1)$ per-node throughput is shown in this work to achieve a delay of $\Theta(n \log n)$. The result, thus, suggests that the use of mobility to increase throughput, even slightly, in real-world networks would necessitate an abrupt and very large increase in delay. The same result is shown by the authors to hold both in the case of a “fluid” model for data as well as for a model with fixed packet-sizes. These pioneering works have thus led to the exploration of delay-tolerant networking models as well as other scenarios that exhibit sparse intermittent-connectivity.

In parallel to these efforts on the general-ad-hoc-networking front, the literature of networking research community has also seen a growing interest in the theory of sensor networks [3,8]. Specifically, these works have focused on basic properties such as the connectivity, coverage, MAC-layer capacity, and routing performance of networks with unreliable nodes that gather data in a given deployment area and communicate them in an efficient manner to a base station. These works have benefited from the theory of random graphs, probability, and percolation theory. Here, the properties of a network wherein nodes are subject to random failures, as in practical sensor-networking scenarios, have been analyzed. These models mostly rely on asymptotic analysis, wherein a large number of sensor-devices are deployed in a dense fashion.

Broadly, this work investigates the theoretical aspects of the design and performance of collaborative ad-hoc networks, making use of probabilistic, graph-theoretic, and information-theoretic approaches. This effort is aimed in the direction of incorporating practical models for such networking scenarios, and in studying relevant

metrics such as latency, lifetime, and throughput achieved by such networks under commonly-encountered routing schemes and communication scenarios. This is in contrast to the previous efforts described above that study asymptotic scaling laws under very idealistic assumptions. The work can be divided into two major parts. The first part focuses on understanding the fundamental performance limits of distributed wireless sensor networks, with specific focus on the lifetime and latency issues, under a convergecast communication paradigm. The second part constitutes the major portion of the thesis and concerns with the performance of Sparse Mobile Ad-Hoc Networks (SPMANETs). This part of the work first focuses on a specific class of SPMANETs known as Delay-Tolerant Networks (DTNs) and presents a novel routing scheme based on fountain codes for efficient message delivery. The work then details a novel analytic framework based on queuing theory and Markov-chain analysis, in understanding the performance limits of SPMANETs. This framework is in turn applied to understand the performance of SPMANETs under various scenarios such as unicast, multiple unicast, multi-hop routing, and network-coded multicast.

1.1 Fundamental Performance Limits in Distributed Wireless Sensor Networks

Recent advances in areas such as micro/nanotechnology, telecommunications, networking and computing have enabled in the development of devices which are small, capable of sensing several environment parameters, and which can process and communicate the information to other wireless devices. Such networks have been termed as *Wireless Sensor Networks* or *Distributed Sensor Networks*. It is predicted that sensor devices would tend towards becoming more and more cost-effective, and will have the potential of being deployed in sufficiently large numbers, opening up venues in applications such as environment monitoring, surveillance (military), road traffic monitoring, safety applications such as chemical leakage detection, etc., [9]. The foremost task of the Sensor Network in any of its applications is to jointly collect and

transmit back the sensed data in response to query requests from the base-station node (henceforth referred to as the *sink* node). The information collected is required to be sufficient enough for the sink to reconstruct the required characteristics of the event or phenomenon reliably. However, any application involving Sensor Networks should also take into account the node constraints i.e., limitedness of available energy, computation power, and memory in order to be practically viable. In particular, energy efficiency is very critical in the design of practical wireless sensor networks. Prolonging battery life in sensor nodes is very important because it might be very expensive or even impossible to replace their batteries because of the hostility of the environment which the sensor might be deployed. Hence, wireless sensor networks need to be deployed with very little infrastructure available, and henceforth, they are expected to work for a satisfactory period of time before the network fails as a result of battery power exhaustion. In other words, typical low-cost sensor nodes are limited by their small memories, short battery lives and low computational power [9], making the design of such networks and communication protocols suitable for the same a challenging problem.

1.1.1 Lifetime Optimization in Wireless Sensor Networks

Energy efficiency is a very critical consideration in the design of distributed wireless sensor networks. Existing literature presents us with a range of protocols and strategies for energy conservation in sensor networks. These include exploiting topology knowledge [10–12], exploiting spatio-temporal correlation i.e., data aggregation, and sleep-active duty cycling [13–15] of sensor nodes. Several data aggregation schemes [16–18], have been suggested which are designed to improve energy efficiency by minimizing redundancy in network traffic. It has also been suggested that sleep-active duty cycling can be incorporated in MAC protocols, and several such MAC protocols have been suggested [16–18]. In addition, the idle-listening state

of the sensor is identified as a major source of power depletion in wireless sensor networks [19–21]. Measurements on existing sensor device radios show that the “idle-listening” state consumes nearly the same power as the “receiving” state. In sensor network applications where the traffic load is light to moderate most of the time, it is therefore desirable to turn off the radio when a node does not participate in any data delivery. Thus [19] and [22] introduce medium access protocols wherein synchronized periodic duty cycling of the sensor nodes is proposed as a mechanism to minimize idle listening radio power consumption, wherein each node follows a periodic cycle of active/sleep radio states. During the sleep period, the radio is completely turned off, and during the active period, the radio is powered back on. However, making sensors go through periodic active/sleep states has one major deficiency - a tradeoff between energy efficiency and *latency* [19] arises. Hence, it is important to estimate the minimum sleep-to-active duty cycle ratio required to ensure that all sensed data is transmitted to the base station (called the sink node). This is useful in scenarios where it is enough to ensure that information about sensed data is able to reach the base station (sink node) within a bounded delay. It turns out that there exists a threshold on the duty cycle (or the probability of being in the active state) beyond which the data delivery delay associated with the network would increase indefinitely in time, making the network useless. Hence, there exists an upper bound on the achievable lifetime of the network. An equally important question is whether we can employ sleep scheduling to minimize the variation in the power consumption of the nodes throughout the network. The latter would ensure that the lifetime of the network is maximized by avoiding network resource wastage.

In this work, we seek to investigate the the problem using a general framework. We address the problem of obtaining bounds on the lifetime and the sleeping probability of nodes that is valid both for event occurrence detection and continuous monitoring type of applications. We show how spatial distribution of nodes plays a very important role

in achieving energy efficiency. In particular, we show that with uniform distribution of nodes, the nodes close to the base station always deplete their energy quickly, no matter what sleep scheduling is used. Hence in this case, the resources in the network are poorly utilized, no matter what networking protocols we employ. We show that by deploying nodes non-uniformly, one can achieve much better energy efficiency in the sensor network. Moreover, we find that a completely asynchronous, independent sleep scheduling scheme is incapable of achieving better lifetime due to channel interference. We show that using an asynchronous quasi-Markov sleep scheduling scheme which uses a low power auxiliary radio, one can amortize node energy consumption throughout the network. Further, we show that there exists a particular deployment scheme which when employed with a particular sleep scheduling scheme would give rise to uniform energy depletion throughout the network, as well as maximum lifetime, while maintaining the data delivery delay bounded. We also show how this could be used in scaling (i.e., adding more nodes to) a distributed sensor network judiciously to achieve very good gains in the network lifetime.

1.1.2 Latency Analysis for Wireless Sensor Networks

In many sensor-networking applications such as military surveillance, chemical hazard detection, etc., *latency* is a critical performance measure, in addition to energy efficiency and lifetime. Latency in a wireless sensor network is defined as the mean time taken for the information about an event occurrence to reach the sink node. This is particularly true in the case of sensor-actor networks since the actors need to function in a timely fashion, which in turn depends on the timely delivery of information by sensor nodes. It is well-known in literature that energy efficiency can only be improved at the cost of the *network latency*, which is the average time taken for information about an event occurrence in the network to reach the sink. It has been

clearly shown [23] that improvement in energy efficiency through data aggregation increases the latency in the network. Duty cycling can be achieved without paying much price in terms of latency, but this requires Time Division Multiple Access (TDMA). However, TDMA schemes are unattractive in the design of sensor networks since they require each sensor node to synchronize its clock with respect to its neighbors, which is hard to achieve and is undesirable because of the additional communication overhead incurred periodically. Adopting a decentralized independent periodic sleep-active duty cycling of sensor nodes means that a node which has a packet to forward has to wait till a subsequent node wakes up. In this work, we provide a comprehensive analytical approach leading to formulations which enable in reliably predicting the latency of distributed sensor networks as a function of several network parameters, namely the network size, duty cycle ratio, event location, incoming data rate generated by the events, and wireless bandwidth. Our approach shall involve results from queuing, graph theory and probabilistic methods.

Latency-related issues in the performance of distributed sensor networks have been recognized in the past. Several authors have described the energy-latency tradeoff issues in specific sensor network scenarios [23–25]. In [23], the authors have studied the problem of scheduling packet transmission for data gathering in sensor networks. They have focussed on the study of energy-latency tradeoffs in wireless sensor networks in such a scenario. Using “aggregation trees” to model the packet flow in multiple source-single sink communication paradigm, they deal with the problem of minimizing the total energy expenditure in the network subject to latency constraints. A numerical algorithm for the exact optimal solution, and a dynamic programming-based approximation algorithm are obtained. In [24], the authors consider the problem of analyzing the tradeoff between energy efficiency and propagation delay, specifically during the deployment phase. They use an analytic model which enables in comparing the tradeoffs for different deployment-phase protocol scenarios,

and present two novel algorithms which outperform existing ones. Analytical modeling of latency has also been considered in the past [26, 27]. In [26], the authors deal with the specific problem of modeling the detection latency (detecting the presence or absence of a specific target) based on collaborative sensing with uncoordinated node mobility. They then analyze the tradeoff between the number of nodes and detection latency in such networks. They formulate the improvement in detection latency for mobile networks as opposed to stationary networks. Perhaps [27] is the closest in existing literature to our own, which analyzes latency in sensor networks with a more generalized setting. The authors in [27] consider the problem of analyzing the latency for information propagation in sensor networks where the nodes go to sleep and wake up independently. They obtain bounds on the time elapsed between the sensing of an incoming event and the time at which such information reaches the sink. The authors model the sleep/active duty cycling of the sensor nodes as a “Blinking Poisson” Boolean model. The results then follow from first passage percolation theory. More specifically, the authors show that messages sent thus reach the sink with an asymptotic speed, which does not depend on the random location of the sensor nodes. However, this analysis is suitable only for networks with light traffic load, since the authors do not consider the effects of interference, congestion, contention etc., which are more relevant in networks with heavy traffic. Further, latency also depends heavily on the routing model. The authors in [27] consider broadcasting scenario, since the results are obtained from Continuum Growth Model, which is used for modeling infection spreading. In our work, we consider the problem of analyzing packet propagation delay in sensor networks, with a broader setting. We consider sensor networks in which multiple nodes collaboratively collect data and transmit information to the sink, called “converge-cast” paradigm. Our model is suitable for networks which can be either lightly, moderately or heavily loaded. We show that for networks with moderate or heavy traffic, interference, congestion and buffering leads

to non-uniform asymptotic speed of packet transmission - i.e., packets generated at different points in the network reach the sink with varying speeds. We further analytically study the effects of interference, incoming data rate and network size on the latency performance, which to our knowledge has not been done in the past for sensor networks. Further, our formulations also show that energy-delay tradeoffs are skewed in networks which are moderately or heavily loaded. Initially, we consider the problem of bounding the latency for large grid-based sensor networks. Grid-based sensor deployments have been studied in [4, 28]. We show that for such networks, an analytic closed form expression can be obtained for latency. We further provide a detailed extension of our analysis to uniform random node deployment.

In our work, we provide an analytical formulation for estimating the latency in sensor networks, and study the effects of various network parameters such as node density, incoming data rate, wireless bandwidth, duty cycle ratio etc., on latency. Our formulations can be employed to obtain fair estimates of network performance, which can be used in practical sensor network design and deployment. Further, our model also shows fundamental limits related to latency in converge-cast communication.

1.2 Performance Characterization of Sparse Mobile Ad-Hoc Networks

Sparse Mobile Ad-Hoc Networking (SPMANET) constitutes a new class of ad-hoc networking architecture that has drawn much attention in the field of wireless networks recently. Often, communication devices are deployed with very little backbone support and exchange information in a collaborative fashion. Additionally in certain scenarios, geographic separation between nodes may be too large, leading to sporadic connectivity. Such networks are often characterized by the lack of acknowledgement messages and by large end-to-end delays.

Conventional models of Mobile Ad-hoc Networks (MANETs) rely on the existence

of end-to-end paths between the source and destination nodes in spite of node mobility. However in SPMANETs, multi-hop paths through which information is sent evolve in space and time. Such special constraints posed by the latter make traditional communication protocols inefficient, and new ones are required in their place. For example, while certain routing schemes designed for general ad-hoc networks such as Dynamic Source Routing [29] fail in such networks, several efficient schemes tailored to these networks have been devised in the past [30–32]. These schemes use the *store, carry, and forward* paradigm (also known as “mobility-assisted routing”) for message delivery, wherein the source node opportunistically transmits packets (intended for a specific destination) to any other node that it comes in contact with, and relies on the mobility of these “relay” nodes to transmit them to the intended destination. As a result, end-to-end paths are created by node mobility in a continuous space-time evolution of the connectivity graph. In this paradigm, communication relies on the mobility of nodes, rather than on the existence of contemporaneous next-hop links leading to an end-to-end path. Such infrastructure-less networks often occur in applications such as wildlife and habitat management [33], defense networks, vehicular networks, and in networks providing cheap, basic internet connectivity to rural areas in developing nations. Depending upon the application context, they may also be known as Delay Tolerant Networks (DTNs). In this part of the thesis, we turn our attention to this class of networks. In the first section, we focus on the design of reliable, efficient message-delivery schemes in Delay-Tolerant Networks. In the second portion, we devote the discussion to obtain a generalized framework for the analysis the SPMANET-class of networks, aimed at modeling the throughput capacity under a variety communication scenarios.

1.2.1 Markovian Approach for Performance Analysis of SPMANETs

The growing interest in the SPMANET networking paradigm has led to the need for mathematical models to characterize their performance behavior. A complete understanding of this new class of networks requires answering the following questions: (1) What are the bounds/estimates of the performance measures (such as latency, capacity, energy efficiency, reliability) for various routing protocols, including network-coding schemes? (2) What are the tradeoffs among these performance measures? (3) How does performance vary in such networks, across different communication scenarios such as unicast, multicast, etc? (4) How does performance scale as a function of network parameters? (5) How does mobility impact performance, or what parameters of mobility affect performance and to what extent?, and (6) To what extent does resource management play a role in performance? To answer the above questions, one needs a rigorous analytical framework for SPMANETs. Thus, the primary focus of this research is to provide such an analytical framework which helps us to design, evaluate, and optimize communication protocols suitable for SPMANETs.

Existing research in SPMANETs and DTNs either oversimplify or narrow down the scope of the problem due to the following two reasons: (1) There has been a tendency in the community to analyze such networks through simple, independent contact-based models such as the Poisson-process model in place of real-world mobility models, due to the former's analytical tractability. Though the Poisson-contact model may work fairly well under very limited scenarios, it is very inaccurate in many cases since it ignores spatio-temporal correlations between contact pairs. (2) The analysis of SPMANETs with practical constraints such as finite available memory is a challenging task. Hence, most of the existing theoretical work in the context of SPMANETs have neglected this key networking consideration. As we will show in this portion of the thesis, finite-buffer effects impact network performance severely, in scenarios wherein the memory available for assisting other nodes' communication is limited.

The significant contributions of this work are as follows:

- We provide a generalized framework for the analysis of SPMANETs under practical considerations such as finite node-memory, unpredictable mobility, channel interference, and node-to-node contention.
- The framework that we provide is valid for a broad class of mobility models that exhibit statistical time-stationarity. This encompasses most of the stochastic mobility models used in the analysis of SPMANET routing protocols, including random walk, random-waypoint, diffusion-based models, etc.
- Our solution provides the means for accurately analyzing networks that demonstrate complex interdependencies and dynamics, by presenting a novel embedded Markov-chain approach. We thus provide a queuing-theoretic approach to the problem of analyzing such networks.
- We apply the above framework to analyze and understand the performance of a few common scenarios, namely simple unicast (two-hop and multi-hop routing), multiple unicast, and network-coded multicast.

**MODELING AND ANALYSIS OF THE PERFORMANCE
OF COLLABORATIVE WIRELESS AD-HOC
NETWORKS: AN INFORMATION-THEORETIC
PERSPECTIVE**

VOLUME I

by

Ramanan Subramanian

CHAPTER II

LIFETIME MAXIMIZATION IN DISTRIBUTED WIRELESS SENSOR NETWORKS

Distributed Wireless Sensor Networks, typically consisting of small low-power sensing devices, are often deployed in applications such as habitat monitoring, hazard detection, and military surveillance. In such scenarios, it is often undesirable or infeasible to periodically maintain the devices (sensing nodes) constituting the network. To complicate this issue further, these devices often need to be deployed in large numbers. Hence, they are often designed with very low computational and communication capabilities, and with very low battery lifetimes. As a result, the communication protocols designed for such networks need to take into consideration the precious little energy resource in the network. Typically, periodic sleep/active duty-cycling of nodes is employed in order to conserve idle-time energy consumption. Moreover, wireless sensor networks are typically deployed to sense, collect, and communicate data in a decentralized manner, and to communicate the same to a centralized base-station node (called the “sink” node). This is in stark contrast to general ad-hoc networks (such as WiFi) in which communication can occur between any arbitrary pair of nodes. A direct implication of this design feature is that nodes close to the sink, which are critical to maintain network connectivity, are often depleted of their energy-resources more quickly than the rest of the network. As a result, efficient sensor-network design needs to address the issue of distributing energy-depletion across the entire network in order to maximize the lifetime. Additionally, sleep scheduling should also take into consideration the minimum utility expected out of the network, so that the network can accomplish the given task in spite of the duty-cycling.

This chapter aims at establishing tight analytical bounds on the sleeping probabilities of nodes and on the achievable lifetime of wireless sensor networks in a generic setting, taking into consideration the issues described above. Bounds on the sleeping probability need to be satisfied for proper network functionality. Further, an energy efficient deployment scheme is suggested wherein the node power depletion is fairly uniformly distributed throughout the network. This scheme makes use of the availability of low power auxiliary channel listening radio. Using a combined deployment and sleep-scheduling approach, we have shown that an improvement in lifetime by a factor of $O(\sqrt{\frac{n}{\log n}})$ over uniform distribution of nodes is achievable, where n is the number of nodes in the network. We also show that the throughput capacity of the network is also improved by the same factor. We show also that the maximum lifetime of the network is bounded above by $O\left(\frac{n^{3/2}}{\sqrt{\log n}}\right)$. Further, the accuracy of our analysis is verified by the simulation results presented. The results pertaining to this chapter are also available in our papers [34, 35].

The rest of the chapter is organized as follows. We first present the assumptions and network models in Section 2.1. In Section 2.2, we show that a purely asynchronous scheme cannot improve the network lifetime. In Section 2.3, we propose a combined deployment and sleep scheduling scheme with which energy consumption can be amortized in the network. We further derive a universal bound on the achievable lifetime of the network. In Section 2.4 we present some of the simulation results which show the validity and accuracy of our analysis. We finally summarize our results with a short discussion in Section 2.6.

2.1 Assumptions and Notations

Let sensor nodes be randomly deployed in a circular region of unit radius, and let a sink node be placed at the center as shown in Fig. 1. Let us assume that a certain physical phenomenon is sensed by the sensors in the network and all events are

reported to the sink. The notations used in our work are explained as below:

Let the total number of Sensor Nodes in the network be n which is large (like 1000's of nodes). Let $r(n)$ be the communication radius of each sensor i.e., a communication link exists between two nodes if the distance between them is less than or equal to $r(n)$. Let us assume that the network has enough nodes as described in [36] and [3]. Let the energy consumption rates for the radio be α_t *Joules/bit* in the transmit mode and let the power consumption be α_i during the radio idle state. The power consumption rate α_i is due to the power spent for keeping the radio powered on. This is independent of the state of the radio - whether it is transmitting or idle. The parameter α_t is associated with the signal power that the node generates during transmission. During the "sleep" state wherein the transmitter and receiver circuitry are switched off, the power consumption be β . Since the radio takes most of the power in the sensor node, β has a much lesser order of magnitude than α_i or α_t . Let the link bandwidth in the wireless channel be W *bits/sec* for a pair of nodes sending and receiving data. It is assumed that the network continuously monitors a physical phenomenon (or reports events which occur as a random process) which generates data in the network at the rate of λ bits per second per unit area in the network.

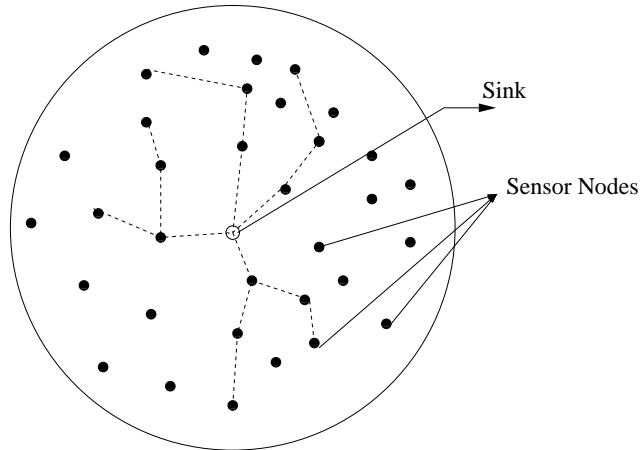


Figure 1: Distributed sensor network controlled by base station

Further, we use a model similar to the *Protocol Model* used in [37] to describe

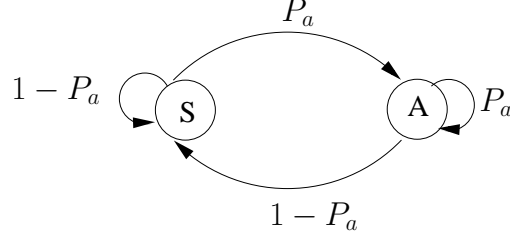


Figure 2: State diagram for asynchronous sleep scheduling

interference-free communication. For successful communication between a pair of nodes, this requires that all nodes within a radius of $(1 + \Delta) r(n)$ must remain silent. Here, the quantity Δ is associated with the tolerable Signal to Interference Ratio in the network. Finally we describe the asynchronous sleep scheduling scheme used in our approach. A sensor node at a distance of ρ from the sink node chooses a quantity called the “wake up transition probability” $P_a(\rho)$ to be determined later. We assume a Markovian model for the sleep scheduling scheme, wherein sensors operate between two states, “Sleep” and “Active”. The transition probabilities for these two states are $1 - P_a(\rho)$ and $P_a(\rho)$ respectively. This is shown in Fig. 2.

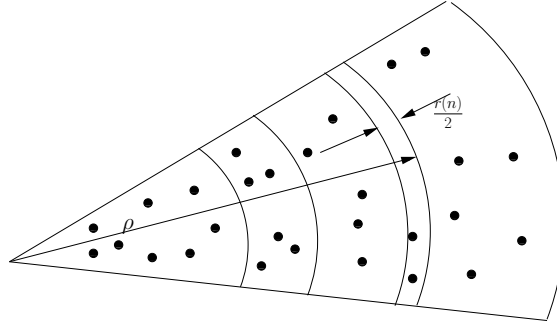


Figure 3: Calculation of node traffic as a function of distance

In addition., we assume that each node also employs distributed source coding so that the data rate transmitted by each node needs to be $\frac{\lambda}{D(\rho)}$ where $D(\rho)$ is the density of nodes at ρ .

2.2 Preliminary Analysis

Consider a node X situated at a distance of ρ from the sink. We divide up the network into hypothetical rings each of which have a thickness of $\frac{r(n)}{2}$. Then, it is very easily seen that any node in a particular ring can communicate with some node in a neighboring ring, given that there are enough nodes in the network to ensure connectivity. It is assumed that the distribution of nodes in the deployment does not change with the orientation at the sink, but may vary as a function of the distance from the sink. In such a case, nodes within the same “ring” as the particular node in consideration (X) handle the incoming traffic from the external “rings” fairly equally among one another. The number of nodes in that particular ring is given by

$$N(\rho) = 2\pi\rho D(\rho) \frac{r(n)}{2} = \pi r(n) \rho D(\rho) \quad (1)$$

where $D(\rho)$ is the node density (*nodes* / m^2) at radial distance ρ from the sink. The traffic generated by any node at a distance ρ' is given by $\frac{\lambda}{D(\rho')}$. Hence, the total traffic at the node X , (which is the sum of the traffic generated by the node X itself and the incoming traffic at the node) is given by

$$\Lambda(\rho) = \frac{\lambda}{D(\rho)} \left(1 + \frac{1}{2\pi\rho\frac{r(n)}{2}} \int_{\rho}^1 \frac{2\pi\rho'}{D(\rho')} D(\rho') d\rho' \right) \quad (2)$$

$$= \frac{\lambda}{D(\rho)} \left(1 + \frac{1 - \rho^2}{\rho r(n)} \right) \quad (3)$$

In order to verify the validity of the above expression, the sensor network traffic was simulated for 1000 nodes deployed uniformly in the network area. The resulting plot in Fig. 4 shows identical behavior with the expression derived in (3).

We first show that the pure Markovian asynchronous sleep-scheduling scheme fails to provide considerable improvement in lifetime no matter how the sensors are deployed in the network. Under such a sleep-scheduling protocol, a node faces interference from other nodes communicating in the vicinity when it wakes up. It can be

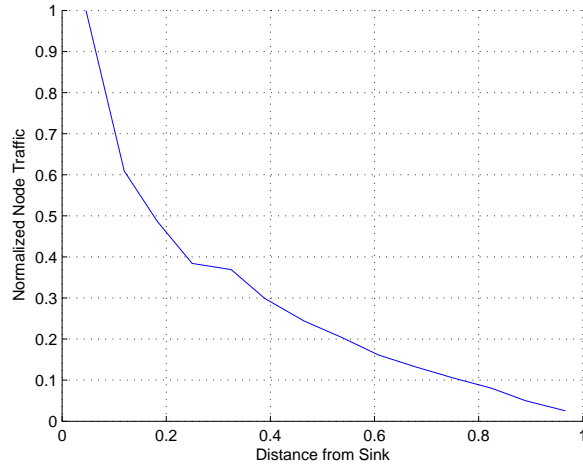


Figure 4: Simulated node traffic behavior as a function of distance

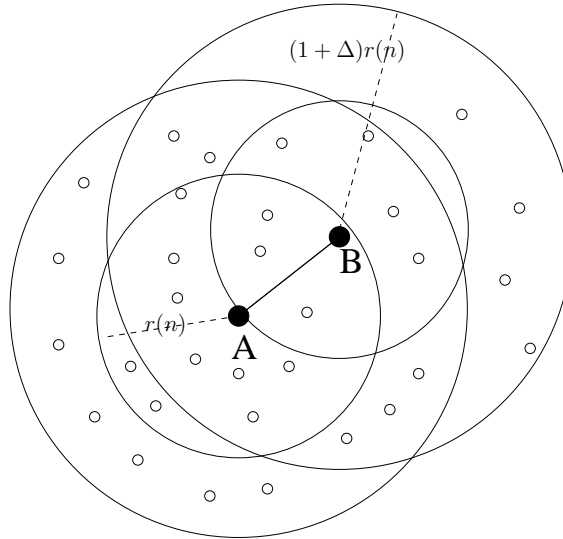


Figure 5: Interference model

easily seen (Fig. 5) that any node shares, on the average, the wireless channel with $D(\rho)\pi r^2(n) \left\{ (1 + \Delta)^2 + \frac{2}{\pi}(1 + \Delta) \right\}$ nodes. Hence the maximum possible data rate that can be transmitted by the node is given by

$$W'(\rho) = \frac{W}{D(\rho)\pi r^2(n) \left\{ (1 + \Delta)^2 + \frac{2}{\pi}(1 + \Delta) \right\}} \text{ bits / sec} \quad (4)$$

Now, the proportion of time spent by the sensor in the active state is given by $\frac{P_a(\rho)}{P_a(\rho)+1-P_a(\rho)} = P_a(\rho)$.

Hence, in order to ensure the feasibility of communication to the sink within a bounded delay, we should satisfy the following fundamental limit for wakeup transition probability:

$$W'(\rho)P_a(\rho) \geq \Lambda(\rho) \text{ bits/sec per node} \quad (5)$$

$$\begin{aligned} \Rightarrow P_a(\rho) &\geq \frac{\lambda}{W} \left(1 + \frac{1 - \rho^2}{\rho r(n)} \right) \\ &\times \pi r^2(n) \left\{ (1 + \Delta)^2 + \frac{2}{\pi}(1 + \Delta) \right\} \end{aligned} \quad (6)$$

Thus, for maximum network lifetime, each sensor chooses its minimum possible wake up probability. i.e.,

$$\begin{aligned} P_a(\rho) &= \pi \frac{\lambda}{W} r^2(n) \left\{ (1 + \Delta)^2 + \frac{2}{\pi}(1 + \Delta) \right\} \\ &\times \left\{ 1 + \frac{(1 - \rho^2)}{\rho r(n)} \right\} \end{aligned} \quad (7)$$

as a function of the distance ρ from the sink. Hence, the power expenditure rate of any node at a distance ρ is given by

$$E(\rho) = \alpha_i P_a(\rho) + \alpha_t \frac{\lambda}{D(\rho)} \left(1 + \frac{1 - \rho^2}{\rho r(n)} \right) \quad (8)$$

We notice that in this case, $P_a(\rho)$ is independent of the distribution $D(\rho)$ and hence the idle/receive time power depletion does not vary with the distribution of nodes in the network. Further, $P_a(\rho)$ increases drastically as we approach the sink.

Since the idle time power depletion is considerable, the power expenditure rate is very high near the sink, which means that sensors near the sink will deplete their energy very fast no matter what the node distribution in the network may be. Further, the nodes farther from the sink will still have considerable power after the nodes close to the sink are depleted. This is also inefficient since there is considerable amount of unused energy resource after the network is dead. This greatly limits the lifetime achievable by the network. This heavy price is due to the interference in the network which handicaps the communication in the case of asynchronous sleeping scheme.

2.3 Proposed solution

2.3.1 Uniform Distribution

Here we propose an asynchronous quasi-markov sleep scheduling scheme, which gives room for improving the lifetime considerably. We slightly modify our scheme so that when a sensor wakes up and the channel is not free, it spends the wakeup time by switching to a low-power auxiliary radio. Sensors with low-power auxiliary radio have been described in [38] and [39] where considerable improvements in energy efficiency have been shown. When the channel is free, the sensor powers up its transmit/receive radio and receives data or transmits data from its buffer. With such a quasi-markov setup, we can see that the sensor can transmit at W bits per second in its wakeup state. This greatly saves on the idle time power consumption. In such a case, we need:

$$\Rightarrow P_a(\rho) \geq \left\{ 1 + \frac{1 - \rho^2}{\rho r(n)} \right\} \frac{\lambda}{W D(\rho)} \quad (9)$$

The node energy consumption rate given by (8) can be derived as:

$$E(\rho) = \left(\frac{\lambda}{W} \alpha_i + \lambda \alpha_t \right) \left\{ 1 + \frac{1 - \rho^2}{\rho r(n)} \right\} \frac{1}{D(\rho)} \quad (10)$$

It can be seen that in the case where we have uniform distribution of nodes, namely, $D(\rho) = \frac{n}{\pi}$, the energy consumption is still very high in the vicinity of the sink compared to the rest of the areas in the network. Hence, even for the “asynchronous quasi-markov” sleep scheduling scheme, uniform distribution of nodes performs poorly. In that case, the maximum battery depletion is given by

$$E(r(n)) = \left(\frac{\lambda}{W} \alpha_i + \lambda \alpha_t \right) \frac{\pi}{n} \frac{1}{r^2(n)} \quad (11)$$

The above expression shows that in order to double the lifetime of the network, one has to double the number of nodes in the network, since inherently, α_i, α_t are directly proportional to $r(n)^2$. Hence the scheme is also unattractive for scaling the lifetime of the network.

Further, the throughput capacity of the sensor network, in terms of *bits per second per unit area*, achieved when $P_{amax} = 1$ is given by $\lambda_{max} = \frac{Wn}{\pi} r^2(n) = O(\log n)$, since $r(n) = O\left(\sqrt{\frac{\log n}{n}}\right)$ as in [3].

2.3.2 Optimum Non-Uniform Distribution

As we have noted above, the most efficient combined-deployment-and sleep-scheduling scheme is one that gives rise to uniform depletion of node battery independent of node location in the network. From (10) it can be easily seen that for uniform energy consumption in the network, the optimum node distribution is

$$D(\rho) = \begin{cases} \frac{1}{E} \left(\frac{\lambda}{W} \alpha_i + \lambda \alpha_t \right) \left\{ 1 + \frac{1-\rho^2}{\rho r(n)} \right\}, \rho \geq \frac{r(n)}{2} \\ \frac{1}{E} \left(\frac{\lambda}{W} \alpha_i + \lambda \alpha_t \right) \frac{4}{r^2(n)}, \rho < \frac{r(n)}{2} \end{cases} \quad (12)$$

where E is the position-independent energy depletion rate of each node. Since the number of nodes in the network is n , $D(\rho)$ satisfies

$$\int_0^1 2\pi\rho' D(\rho') d\rho' = n \quad (13)$$

$$(14)$$

Therefore, we have

$$E = 2\pi\lambda \left(\alpha_t + \frac{\alpha_i}{W} \right) \frac{4}{3nr(n)} \quad (15)$$

In which case, the optimum sleeping probability can be derived as:

$$P_a(\rho) = \frac{8\pi\lambda}{3W} \frac{1}{3nr(n)} \quad (16)$$

From the above equations, one can see that there exists a non-uniform way of deploying nodes according to the distribution $D(\rho)$ as in (12) so that the energy depletion is uniform throughout the network. This gives rise to good improvement in the lifetime as compared to (11). From (11) and (15) one can see that the improvement in the lifetime is by an order of $O(\frac{1}{r(n)})$ or $O(\sqrt{\frac{n}{\log n}})$. Hence, our combined deployment and quasi-markov asynchronous sleeping scheme performs better than both the pure markov asynchronous scheme as well as the uniform distribution scheme by a factor of $O\left(\sqrt{\frac{n}{\log n}}\right)$.

Hence, we also have a bound on the maximum achievable lifetime of the network:

$$T_{Lmax} = O\left(\frac{n^{3/2}}{\sqrt{\log n}}\right) \quad (17)$$

Further, the throughput capacity of the sensor network, in terms of *bits per second per unit area*, λ_{max} achieved for $P_a(\rho) = 1$, in this case is $\lambda_{max} = \frac{3W}{8\pi} nr(n) = O(\sqrt{n \log n})$, which again shows an improvement from the uniform deployment scheme by a factor of $O\left(\sqrt{\frac{n}{\log n}}\right)$.

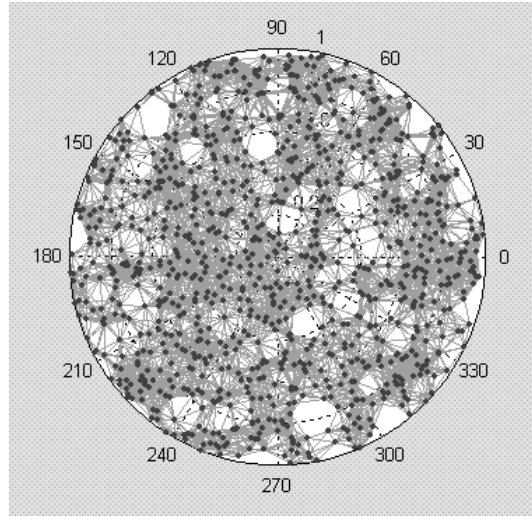


Figure 6: Network with uniform deployment of sensors

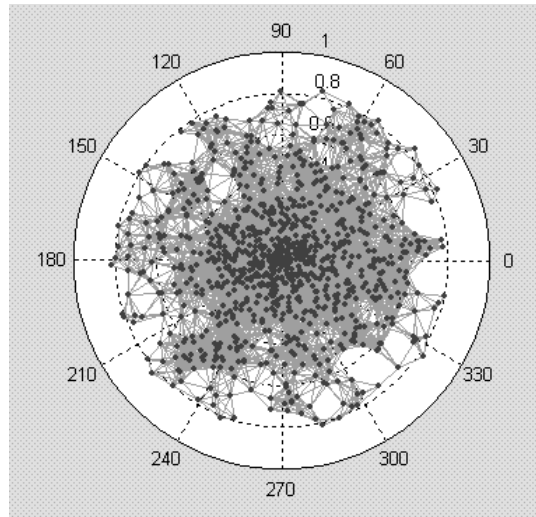


Figure 7: Network with optimal deployment of sensors

2.4 Simulation Results

The performance of the sensor network was simulated in two cases, viz., uniform and non-uniform quasi-markov asynchronous sleep scheduling schemes. The results show that the proposed combined deployment and sleep scheduling scheme outperforms the former considerably by a large factor. In the simulations, it was assumed that 1000 nodes are randomly deployed in a circle of unit radius. The radio communication range of each sensor was assumed to be $r(n) = 0.1$. Each node transmit data at such a rate such that the data density (λ) is 100 kbps per unit area. The sensor network was simulated both for uniform distribution (Fig. 6) as well as for the energy-efficient non-uniform distribution (Fig. 7) of nodes. We have used the Greedy Perimeter Stateless Routing described in [40] as the routing scheme for the sensors. Using this routing scheme and the proposed sleep scheduling scheme, we have obtained the plots of node energy consumption as a function of the node distance in Figs. 8 and 9. These plots clearly show that the node energy consumption is fairly uniform for the non-uniform deployment case and is very uneven, with maximum depletion near the sink, in the case of uniform deployment of sensors.

Further, these plots also show the validity of our analytical results, as the energy consumption rates as a function of node distance matches with the analytical results obtained in the previous section, as can be seen from the plots of the expressions derived in 2.2 and 2.3 in Figs. 8 and 9. Further, our results show considerable energy improvements. The values of the power consumptions achieved by our simulations are summarized below:

Table 1: Simulation Results

| Parameters | Uniform Deployment | Nonuniform |
|----------------------|--------------------|------------|
| $n = 1000; r = 0.15$ | 0.1767mW | 0.0224mW |
| $n = 2000; r = 0.10$ | 0.0753mW | 0.0070mW |

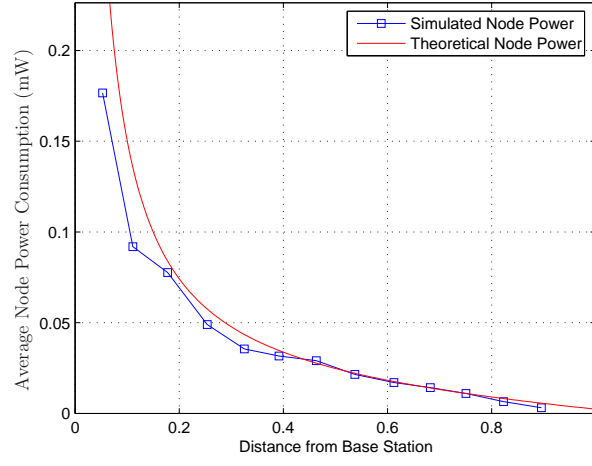


Figure 8: Plot of Node Power Expenditure (1000 nodes) (Uniform Deployment)

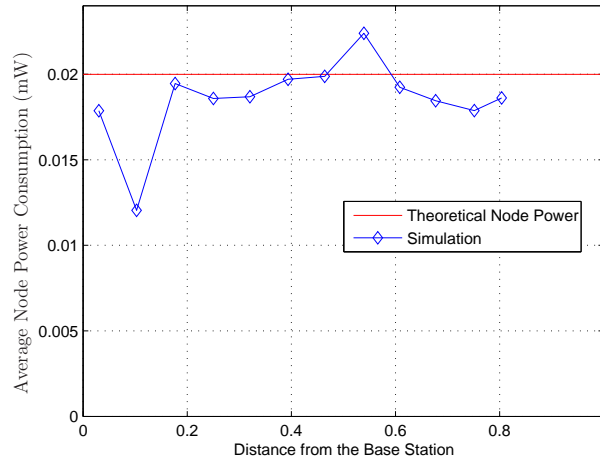


Figure 9: Plot of Node Power Expenditure (1000 nodes)(Optimum Non-Uniform Deployment)

There is a slight downside in the proposed scheme since the node density is not good enough beyond $\rho \approx 0.7$, and hence the coverage of the network is reduced. However, the rest of the network is fully connected and covered. Hence, one could scale the radio range to provide full coverage in the network. This means the achievable lifetime is reduced by a factor of about two. However, this does not change the order of magnitude of the results obtained in the previous section, and it still outperforms any other scheme since the number of nodes is large.

2.5 *Significance of our Results*

The significant implications of our work to the design of practical wireless sensor networks are as follows.

- *Lifetime of Sensor Networks in pure Markovian asynchronous scheme:* Assuming that the sleep-active duty cycling of sensor nodes can be modeled as a Markov process, we show that such a purely asynchronous probabilistic scheme does not allow for considerable improvements in the lifetime of the network. In such a case, we show that the energy depletion in the vicinity of the sink is maximum, no matter how sensors are deployed in the network geographically.
- *Quasi-Markovian Asynchronous sleep scheduling scheme:* We suggest and prove that using the novel “low-power wake up radio” sensor suggested in [39] and [38], one can devise a quasi-Markovian asynchronous duty cycling scheme wherein the sensor nodes deterministically switch to the low-power auxiliary radio when it detects that the channel is not free on wake up, and powers up its radio when the channel is free.
- *Energy-efficient deployment of sensor nodes:* We show that for conventional uniform deployment, even while employing a “low-power wake up radio” at the sensor nodes, energy depletion in the network is quite rapid. We show

that using the wake up radio strategy for sleep-scheduling, one could obtain a fairly good energy-efficient deployment scheme which amortizes the energy depletion throughout the network. In addition to being much more energy efficient compared to the uniform deployment case, the above scheme also makes optimal use of the energy resources in the network before the network itself becomes dysfunctional due to energy depletion.

- *Maximum achievable lifetime of a Sensor Network:* Independent of deployment, routing, MAC protocol, and sleep scheduling schemes employed, we obtain upper bounds on the achievable lifetime of Wireless Sensor Networks. We further show that the combined deployment and sleep-scheduling scheme described above achieves the bound within a difference of a constant factor. We also show that our scheme is excellent for scaling the lifetime by increasing the number of nodes, since the number of additional nodes required would be very small compared to the factor of improvement in the lifetime.

2.6 Summary

We have obtained the fundamental limits on the sleep duty cycling of sensor nodes for energy-efficient operation of the network. We have also obtained upper bounds on the achievable lifetime of the sensor network, independent of routing, sleep scheduling and deployment schemes. We have proposed a novel approach to energy-efficient asynchronous sleep scheduling using non-uniform deployment of sensors. We have shown that using a judicious combined deployment and sleep scheduling scheme, one can obtain significant improvements in the network lifetime.

CHAPTER III

LATENCY ANALYSIS FOR CONVERGECAST COMMUNICATION IN A DISTRIBUTED SENSOR NETWORK

The previous chapter focused on energy-efficiency considerations in a typical distributed-sensor-network scenario. However, this study did not consider the problem of increase in latency as a result of duty cycling aimed at enhancing the network lifetime. Hence, there is a natural tradeoff between the design goals of lifetime and allowable latency in a distributed sensor network. In this chapter, we aim at studying this tradeoff as a function of several design parameters, namely the network size, duty cycle ratio, event location, incoming data rate generated by the events, and wireless bandwidth. Further, the scalability of a given sleep-scheduling protocol needs to be studied, as it would be undesirable for the tradeoff to be drastically severe when a given design parameter is varied. Ideally, a given scheme should be insensitive to variations in the design parameters so that it can be made globally applicable.

In this work, we consider a sensor network model in which nodes detect an event jointly and transmit the related event information to a central base station. We address the problem of finding analytical estimates for the average latency incurred in reporting the sensed event to the sink (called “convergecast” communication). We consider an application model wherein events occur as a Poisson process in time, with the location of the event being uniform throughout the deployment area. We further assume that sensor nodes go to sleep and wake up independently as in a Markov process. We obtain analytical bounds for the average latency for reporting an event, as a function of the location of the event with respect to the sink, as well as a function

of several network parameters such as network size, radio range, data rate, duty cycle ratio and wireless bandwidth. Detailed analysis is provided for the case when nodes are deployed in a regular grid. We also provide similar analysis for the case when nodes are deployed according to a uniform random process. The analytical expressions obtained are validated with simulations in each case. The results pertaining to this chapter are also available in our papers [41, 42].

The important contributions of this work are as follows:

- *Asynchronous Sleep Scheduling:* In our work, we avoid the need for the nodes to synchronize with each other in the network. Further, the nodes choose their sleeping probabilities independent of others. Hence, our scheme is easy to implement and does not require complicated synchronization protocols which lead to undesirable overhead.
- *Application to Sleep Scheduling Schemes:* We obtain bounds on the sleep-active duty-cycle ratio for sensor nodes in the network. Since our assumptions and modeling are fairly generic, these bounds need to be satisfied when one incorporates sleep-active duty-cycling in sensor nodes. Hence, our results can stand as a good universal reference point for choosing sleep scheduling schemes for sensor nodes.
- *Adaptive Sleeping Scheme for Dynamic Environments:* Very often, sensor networks are deployed in dynamic environments where events may not occur with any regularity. In such a case, our analytical results show how a node could adaptively choose its duty cycle based on the current traffic level conditions.
- *Application to Clustering-based Sensor Networks:* Our results can be easily extended to the case of Clustering-Based Networks. In such a case, the results obtained on the node density apply for the distribution of cluster-heads in the

network. Hence, our results can be extended to obtain energy-efficient clustering schemes.

The rest of the chapter has been organized as follows. In Section 3.1, we present our approach for the analysis of latency in grid sensor networks. In Section 3.2, we discuss the validity of our results using sensor network simulations. In Section 3.3, we discuss an extension of our approach to analyze latency in sensor networks involving uniform random deployment of nodes. We present a few simulation results to validate the analysis for uniform random deployment in Section 3.4. We finally conclude this chapter in Section 3.6

3.1 Analysis of Network Latency for Grid Sensor Networks

Consider a network of sensor nodes arranged in a square grid like fashion, where the size of the grid is $m \times m$, (assume that m is odd). The deployment region is a square of unit area. The sink node is located at the center of the grid, and all communication is directed towards this node. Hence there are $m^2 - 1$ sensor nodes in the network. Let $(0, 0)$ represent the sink node. Then sensor nodes are located at $(\pm i, \pm j)$ where $i, j \in \{1, 2, \dots, m-1\}$. Two sensors can communicate with each other when they are within a distance of $\frac{1}{m-1}$, i.e., there are no “diagonal” links in the grid network.

Now we describe the data model used in our analysis. We use a poisson process model for the event occurrence. Hence, it is assumed that at each node, events occur with a frequency of λ_0 per second and each event generates a single packet consisting of B bits of information which needs to be transmitted to the sink. For now, we assume that there is no spatial correlation of events generated by two different nodes. It is possible to extend our analysis further to correlated data, by dividing the network into clusters based on correlation regions. We assume that nodes employ geographic routing to forward gathered data hop-by-hop to the sink i.e., every node that needs to transmit a packet forwards it to one of its neighbors that is closer to the sink

(by hop count) than itself. It is assumed that the wireless channel has a bandwidth of W bits per second. The interference model considered is such that when two sensors participate in communication, all other neighbors of both the transmitter and receiver are to remain silent. It is also assumed that in order to conserve battery power, each node employs an independent sleep-active duty cycling according to a stationary Markov process. The reason for this assumption will be clear when we assume that packet forwarding times are exponentially distributed. In short, it can be assumed that the transition probabilities for sleep-to-active and active-to-sleep states are respectively P_a and $1 - P_a$, respectively.

The average node traffic can be formulated as follows. It can be easily seen that from our network setup, nodes which are the same number of hops from the sink have the same average traffic. The number of nodes which are at a hop distance of i from the sink can be shown to be $4 \left(\frac{m}{2} - \left| i - \frac{m}{2} \right| \right)$. The number of nodes which are at a hop distance of i or greater can thus be found as $(m^2 - 1) - 4 \sum_{j=1}^{i-1} \left(\frac{m}{2} - \left| j - \frac{m}{2} \right| \right)$. Hence, the average traffic $\lambda(i)$ of a node at a hop level i from the sink can be found as

$$\lambda(i) = \frac{\left(\frac{m^2-1}{2} \right) - m(i-1) + \sum_{j=1}^{i-1} |2j - m|}{m - |2i - m|}$$

Every packet that arrives at a node is queued up in the node's buffer and scheduled for forwarding to the sink. The packet arrival rate for a node at a hop distance i is given precisely by $\lambda(i)$. We need to find out the service delay for any packet at that particular node, and to do so, we need to find out the service rate $\mu(i)$ at that node. This can be obtained as follows - the channel bandwidth is given by W as mentioned before. It can also be seen that 22 other links can interfere with a single link as shown in Fig. 10. However, assuming m is large, for most links, there is also an alternative next-hop to reach the sink by geographic routing. Hence, effective available bandwidth for most nodes (except fringe nodes and nodes exactly at a

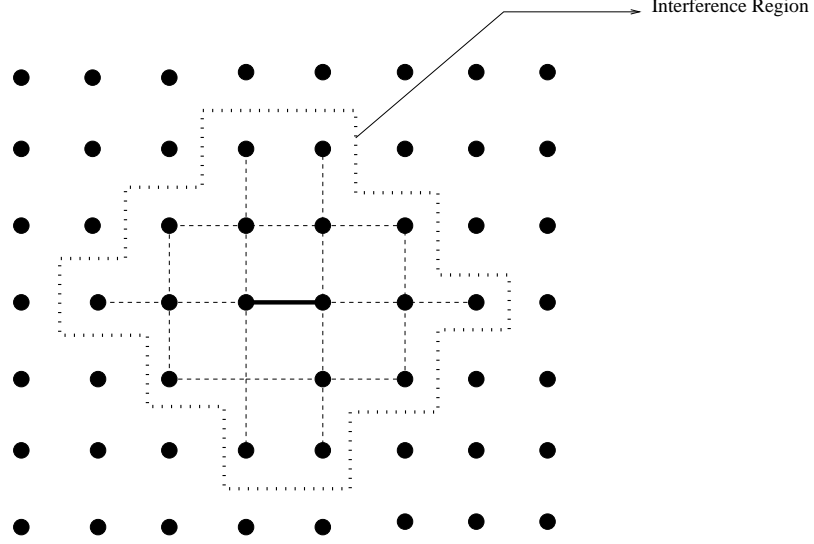


Figure 10: Interference region in the grid network

horizontal/vertical position with respect to the sink node) is given by $\frac{2W}{23B}$. Further, since nodes go through periodic active-sleep duty cycling, the probability that at any given time a link is available for communication is given by $2P_a - P_a^2$. Hence the effective service rate for a node is given by

$$\mu(i) = \frac{2W}{23B} (2P_a - P_a^2)$$

Hence, the expected packet delay at a node i hops away from the sink is given by

$$\tau(i) = \frac{1}{\mu(i)} \frac{1}{\mu(i) - \lambda(i)}$$

This is a result from basic queuing theory ($M/M/1$ queues). However, this needs to be justified since the $M/M/1$ model assumes that the packet arrivals are according to a Poisson process. However, we have only assumed that event occurrence is a Poisson process in time. To show Poisson-like behavior for overall packet arrival at a node, we just need to show that the inter-arrival times are exponentially distributed. We do so by simulation for a couple of cases, since exact analysis is more far-fetched and we do not present it due to space constraints. The plots show that for the Poisson process event model used in this work, the inter-arrival times in fact show exponential

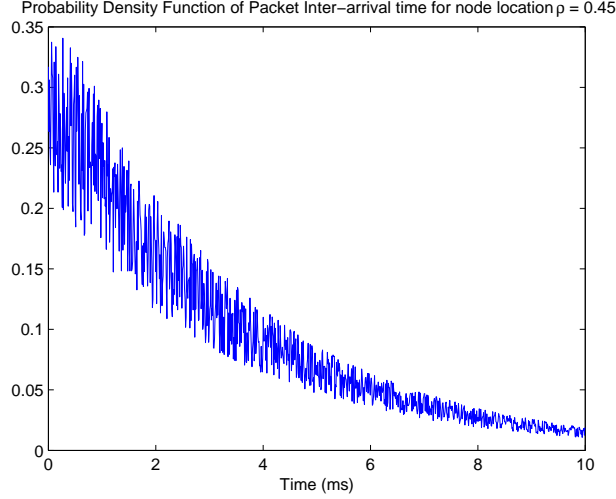


Figure 11: Probability Density Function of inter arrival times for 8 bps source traffic and a different node location

behavior. A plot of the simulated probability density function of inter-arrival times in Fig. 11, averaged over 500 iterations shows that inter-arrival times exhibit exponential behavior, so we can model packet arrivals at any node by a Poisson process, and hence use results from queuing theory.

Hence, the total time taken for a packet to reach from a hop level, say l to the sink can be given by

$$\begin{aligned}
 T(l) &= \sum_{i=1}^l \tau(i) \\
 &= \lambda_0 \frac{23B}{2W(2P_a - P_a^2)} \sum_{i=1}^l \frac{\eta(i)}{\xi(i)}
 \end{aligned}$$

where

$$\begin{aligned}
 \zeta(i) &= \frac{(m^2 - 1)}{2} - m(i - 1) + \sum_{j=1}^{i-1} |2j - m| \\
 \xi(i) &= \frac{2W}{23B} (2P_a - P_a^2) (m - |2i - m|) \\
 &\quad - \frac{(m^2 - 1)}{2} - m(i - 1) + \sum_{j=1}^{i-1} |2j - m|
 \end{aligned}$$

The above expression is not very convenient and it does not yield an analytically closed form expression for the average delay. This is because the expression for node

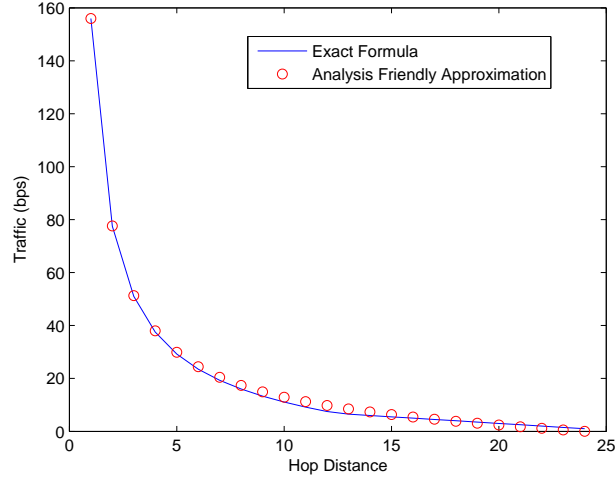


Figure 12: Plot of node traffic vs hop distance for $m = 25$

traffic $\lambda(i)$ is very complicated. For large scale networks, it can be simplified as follows $\lambda(i) = \lambda_0 \frac{m+1}{4} \left\{ \frac{(m-1)^2 - i^2}{(m-1)i} \right\}$. This can be justified from a comparison of the exact expression for the traffic with the approximate expression as in Fig. 12.

Further, since $\lambda(i)$ is monotone decreasing and hence $\tau(i)$ is monotone decreasing, we can bound $T(l)$ as

$$\int_1^l \frac{1}{\mu(x)} \frac{1}{\mu(x) - \lambda(x)} dx \geq T(l) \geq \int_1^l \frac{1}{\mu(x)} \frac{1}{\mu(x-1) - \lambda(x-1)} dx$$

and for large-scale networks,

$$T(l) \approx \int_1^l \frac{1}{\mu(x)} \frac{1}{\mu(x) - \lambda(x)} dx$$

Let us write

$$\begin{aligned} \mu &= \frac{2W}{23B} (2P_a - P_a^2) \\ \beta &= \frac{2(m-1) \left(\mu - \frac{m+1}{4} \lambda_0 \right)}{\lambda_0 (m+1)} \\ \gamma &= \left(\frac{2(m-1)}{\lambda_0 (m+1)} \sqrt{\left(\mu - \lambda_0 \frac{m+1}{4} \right)^2 + \frac{\lambda_0^2}{4} (m+1)^2} \right) \end{aligned}$$

we can analytically evaluate $T(l)$ as

$$\begin{aligned}
T(l) &\approx \frac{1}{\mu} \int_1^l \left\{ \frac{\mu(m-1)x}{\mu(m-1)x - \lambda_0 \frac{m+1}{4}} - 1 \right\} dx \\
&\times ((m-1)^2 + (m-1)x - x^2) \\
&= \frac{2(m-1)}{\lambda_0(m+1)} \log \left(\frac{(l+\beta)^2 - \gamma^2}{(1+\beta)^2 - \gamma^2} \right) - \frac{23B}{2W(2P_a - P_a^2)} (l-1) \\
&+ \frac{8(m-1)^2 \left(\mu - \frac{m+1}{4} \lambda_0 \right)}{(m+1)^2 \lambda_0^2} \left\{ \log \left(\frac{1+\beta+\gamma}{1+\beta-\gamma} \right) \log \left(\frac{l+\beta+\gamma}{l+\beta-\gamma} \right) \right\} \quad (18)
\end{aligned}$$

Thus the above expression estimates the latency in grid sensor networks employing sleep scheduling to save energy, as a function of various network parameters. From the above analysis, one can see that as a packet approaches the sink, its propagation towards destination gets considerably “slowed down” due to congestion and buffering effects. Hence, in contrast with [27], packets generated at different locations do not necessarily approach the sink at the same speed. Further, it can also be shown that as we scale the network, the average increase in latency is worse than linear. Hence, we pay a heavier than expected price in terms of latency when the number of nodes is increased to include more coverage area. We further present plots of the analytical expressions, in order to clearly understand the trade-offs between latency and other performance metrics in the network. These plots have been obtained for moderately to heavily loaded networks, since the effects of the parameters such as the duty cycle ratio (P_a), bandwidth (W) and input data rate (λ_0) would be weaker and less pronounced in lightly loaded network. Figure 13 shows a plot of the latency as a function of the idle state energy expenditure. Trends observed in Fig. 13 show that we pay a heavy price on latency for comparatively smaller gain in energy efficiency. The effect of the input data rate (λ_0) and the wireless channel bandwidth (W) are shown

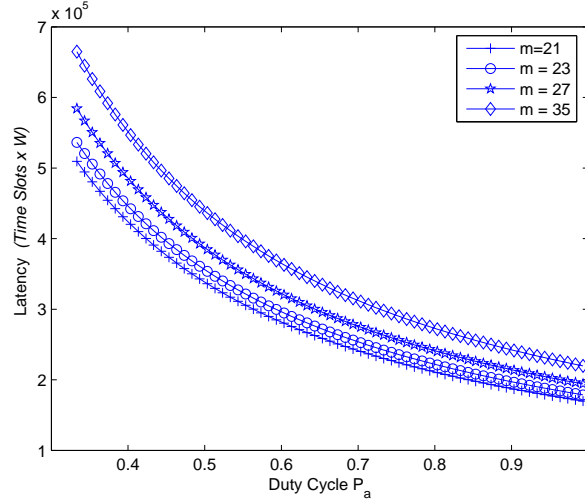


Figure 13: Latency (Time Slots $\times W$) as a function of the duty cycle

in Fig. 14. This plot shows that latency increases exponentially as we approach the network capacity, but decreases with increase in the wireless bandwidth, as expected.

3.2 Simulation Results

In this section, we verify the analytical results in the previous sections through extensive simulations on a MATLAB platform. We consider a grid network consisting of 624 sensor nodes and a sink at the center of a 25×25 grid (i.e., $m = 25$), deployed in a square region of unit length, and nodes within a distance of $(\frac{1}{m-1})$ can communicate with each other. Each node was simulated to generate packets as a spatially uniform, temporally Poisson process, as described previously. The wireless bandwidth was chosen to be 128 kbps. The packets, of length 32 bits each, generated at each node is transferred hop by hop to the sink. Then, the average delay for each hop level is computed over 2000 iterations, and an overall average latency for the entire network was obtained. The simulations were run with varying input data rate (λ_0) and duty cycle (P_a). Our results from the simulations are plotted, along with the corresponding analytical expressions obtained previously (within a tolerable error range), in Fig. 16 and Fig. 17. One can also see that the effects of various parameters

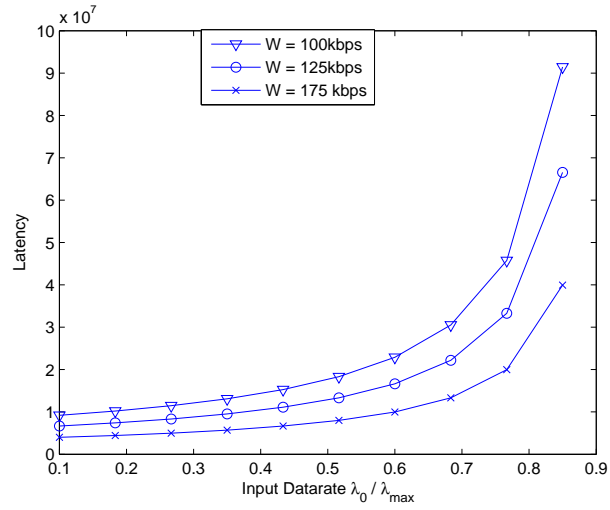


Figure 14: Plot of latency vs event frequency for different bandwidths

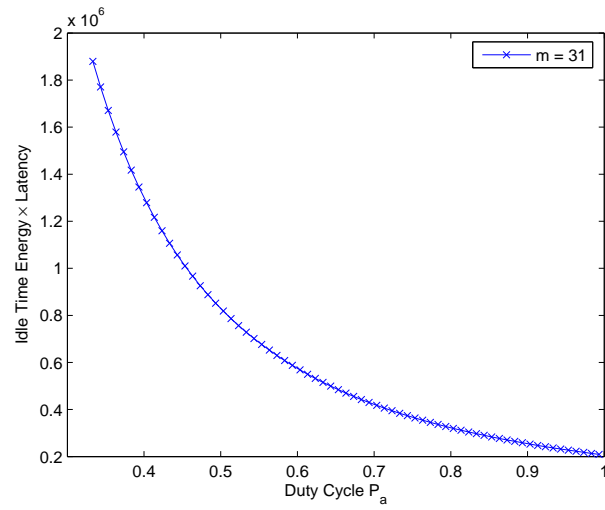


Figure 15: Idle Time Energy-Latency product as a function of duty cycle

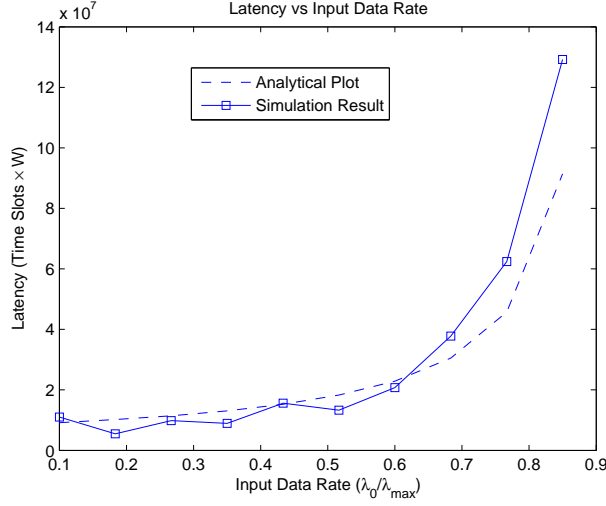


Figure 16: Simulation : Latency vs Input Data Rate

that have been discussed in the previous section prove to be similar in our simulation results. For a duty cycle of 0.4 at each node, simulations were run with varying input data rates, from 10% to 90% of the capacity of the network (53 packets per second, found from separate simulations) under the given scheme. The plot in Fig. 16 shows, as expected, exponential rise in the latency as we approach the capacity limit. For the case in Fig. 17, an input data rate of $\lambda_0 = 30$ packets per second was simulated, and the delay was computed for duty cycles varying from 10% to 90%. Once again, we observe that we pay a heavy price in terms of latency, if we decrease the duty cycle to gain energy efficiency (as duty cycling reduces the idle time energy wastage).

3.3 *Extension to Uniform Random Deployment*

So far, in our analysis, we have considered distributed sensor networks deployed in a grid fashion. However, this is valid only in cases where it is feasible to control the node position. In many practical scenarios such as defence, habitat monitoring etc., the location of the nodes cannot be predetermined. In such applications, the network is deployed by spraying nodes randomly in the deployment area.

In this section, we give a detailed step-by-step analysis of the latency of such

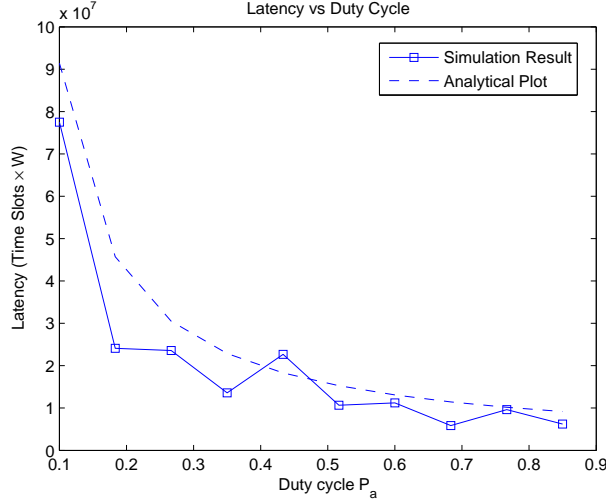


Figure 17: Simulation : Latency vs Duty Cycle

networks. Though a compact formula cannot be extracted, it suffices to say that given the network parameters, our model and analysis can be used to reliably estimate the latency performance of the network and in studying the related tradeoffs, instead of going for computation-intensive simulations.

Let sensors be randomly deployed in a circular region of unit radius according to the uniform distribution, with the sink node located at the center. Suppose a physical phenomenon consisting of events occurring at random (as in Section 3.1) is sensed by the sensors in the network and all events are reported to the sink. Let the total number of Sensor Nodes in the network be n which is large (like 1000's of nodes). Let $r(n)$ be the communication radius of each sensor i.e., a communication link exists between two nodes if the distance between them is less than or equal to $r(n)$. Let us assume that the network has enough nodes as to ensure connectivity. Further, we use a model similar to the *Protocol Model* used in [1, 2] to describe interference-free communication. For successful communication between a pair of nodes, this requires that all nodes within a radius of $(1 + \Delta)r(n)$ must remain silent. Here, the quantity Δ is associated with the tolerable Signal to Interference Ratio in the network. We again assume that nodes go to sleep and wake up as a Markov process,

and do so independently. As before, let the link bandwidth for the wireless channel be W *bits/sec* for a pair of nodes sending and receiving data. It is assumed that the network continuously monitors a physical phenomenon (or reports events which occur as a random process) which generates data in the network at the rate of λ_0 bits per second per unit area in the network.

As before, we employ geographical routing described in [12], excepting that to make analysis tractable, we use a slight modification of the same, which is an approximation to the actual geographic routing scenario - each sensor node tries to forward a packet to a destination closer to the sink than itself, within a field of 120° in its communication range.

Consider a node X located at, say, a distance of ρ from the sink. Let us say that an event occurs, and this node generates a packet to be sent to the sink. Then, the expected delay for the packet can be calculated as:

$$T(\rho) = \sum_{p \in \mathcal{P}} \tilde{T}(p) P(p)$$

where

- \mathcal{P} is the set of all possible paths (subject to the geographical routing constrained discussed previously in this section) from X to the sink node.
- $\tilde{T}(p)$ is the expected delay for the path p
- $P(p)$ is the probability that the routing follows path p

We can rewrite the above as

$$T(\rho) = \sum_{l=1}^{\infty} \tau(l) P_s(l) \tag{19}$$

where $\tau(l)$ is the expected delay of a path with length l and $P_s(l)$ is the *probability mass function* representing the ratio of paths with length l , out of all possible paths

- known as the *path spectrum* of the graph representing the network, as described in [43].

Hence, the problem boils down to computation of path spectrum and the delay for a particular path of length l . We have observed that both can be done for the case of large scale networks, provided that node X is located moderately far from the sink i.e., $\rho \gg r(n)$, so that the hop count to the sink is quite large ($\sim > 20$).

3.3.1 Path Spectrum Estimation

In geographical routing, each hop gets closer and closer to the sink, as shown in Fig. 18. Hence for a packet originating from a node X , let the successive reductions in the distance from sink of each node location in its path (ρ) be denoted as random variables Y_1, Y_2, \dots, Y_l . For $\rho \gg r(n)$, we can in fact model these as i.i.d random variables. The probability density function for each of these variables can be found with a little effort as:

$$f_X(x) = \begin{cases} \frac{6\sqrt{3}}{\pi r^2(n)} x, & 0 \leq x \leq \frac{r(n)}{2} \\ \frac{6}{\pi r^2(n)} \sqrt{r^2(n) - x^2}, & \frac{r(n)}{2} \leq x \leq r(n) \end{cases}$$

$$\text{where } \nu = E[X] = \frac{\sqrt{3}}{\pi} r(n)$$

$$\nu^2 + \sigma^2 = E[X^2] = \frac{6}{\pi} \left[\frac{\pi}{24} + \frac{\sqrt{3}}{32} \right]$$

Then, the probability that the path length for a randomly chosen path is l is given by the probability that both

$$Y_1 + Y_2 + \dots + Y_{l-1} \leq \rho - r(n)$$

$$Y_1 + Y_2 + \dots + Y_l \geq \rho - r(n)$$

writing $Y_1 + Y_2 + \dots + Y_{l-1} = Z$, we can see that by central limit theorem, the density function of random variable Z is approximately gaussian with mean $\nu(l-1)$ and

variance $\sigma\sqrt{l-1}$. Hence, we have

$$f_Z(z) = \frac{1}{\sigma\sqrt{2\pi(l-1)}} e^{-\frac{\{z-(l-1)\nu\}^2}{2(l-1)\sigma^2}} \quad (20)$$

where $f_Z(z)$ is the probability density function of the random variable Z .

Now, one can easily see that the probability that $Z+Y_l \geq \rho-r(n)$ and $Z \leq \rho-r(n)$, which is the same as the mass function $P_s(l)$ can be written as

$$P_s(l) = \frac{1}{\sigma\sqrt{2\pi(l-1)}} \int_{-\infty}^{\rho-r(n)} e^{-\frac{\{z-(l-1)\nu\}^2}{2(l-1)\sigma^2}} (1 - F_X(\rho - r(n) - z)) dz \quad (21)$$

where $F_X(x) = \int_0^y f_X(x)dx$. The above expression for path spectrum can be evaluated in terms of Bessel functions. A simulation of actual hop by hop forwarding of 10000 packets shows that the analytical expression derived thus agrees with the simulation results. The plots for both simulation and analysis are shown in Fig. 19.

3.3.2 Delay Computation for a Given Path Length

We now provide an estimate of the expected delay for a packet originating from node X located at ρ for a given path length l . To do so, as before, we need to find the average traffic function at each node $\lambda(\rho')$ and the average service rate μ' at a given node. The former has been derived for our setting in the previous chapter, and we use the same expression -

$$\lambda(\rho') = \lambda_0 \frac{\pi}{n} \left\{ 1 + \frac{1 - \rho'^2}{r(n)\rho'} \right\} \quad (22)$$

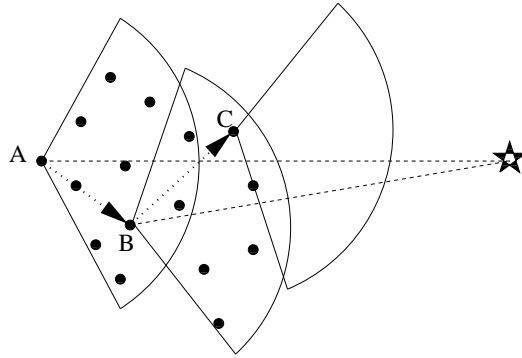


Figure 18: Illustration of Geographical Routing

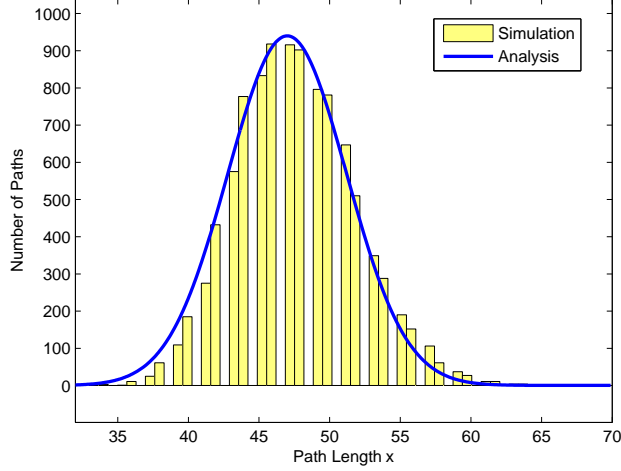


Figure 19: Comparison of Simulated and Estimated Path Spectrum

To find μ' , we note that for the interference model we consider, the area of the *interference region* [1] derived in [34] is given by

$$A = \pi r^2(n) \left\{ (1 + \Delta)^2 + \frac{2}{\pi} (1 + \Delta) \right\} \quad (23)$$

The number of links sharing an interference region is computed as follows - each node has, on an average $nr^2(n)$ neighbors, and there are $\frac{n}{\pi}A$ nodes in the deployment region. Hence, the total number of links is $\frac{1}{2} \frac{n^2}{\pi} r^2(n)A$, after accounting for the fact that the product of the former two counts each link twice. Further, on the average, there are $\frac{1}{3}nr^2(n)$ possible next-hops for each node. Hence, the average service rate at any node is given by

$$\mu' = \frac{2W\pi}{n^2r^2(n)BA} (2P_a - P_a^2) \times \frac{1}{3}nr^2(n) \quad (24)$$

$$= \frac{2W}{3B} \frac{1}{nr^2(n)} \frac{1}{\left\{ (1 + \Delta)^2 + \frac{2}{\pi} (1 + \Delta) \right\}} \quad (25)$$

where B is the packet size in bits. As before, assume that random variables Y_1, Y_2, \dots, Y_l represent the radial improvement in the advance of a packet towards the sink node.

Let

$$g(x) \triangleq \frac{1}{\mu'} \frac{1}{\mu' - \lambda(\rho - x)} \quad (26)$$

Then, the delay encountered by a packet after completing k of its l hops is clearly given by $g(Y_1 + Y_2 + \dots + Y_k)$. Again, we invoke the central limit theorem on Y_1, Y_2, \dots, Y_l . Let ν and σ be defined as in the previous subsection. Then, the expected delay after k hops is given by

$$\tilde{\tau}(k) = \frac{1}{\sigma\sqrt{2\pi k}} \int_{-\infty}^{\infty} g(x) e^{-\frac{(x-k\mu)^2}{2k\sigma^2}} dx \quad (27)$$

Hence, the total delay for l hops is given by

$$\begin{aligned} \tau(l) &= \sum_{k=1}^l \tilde{\tau}(k) \\ &= \sum_{k=1}^l \frac{1}{\sigma\sqrt{2\pi k}} \int_{-\infty}^{\infty} g(x) e^{-\frac{(x-k\mu)^2}{2k\sigma^2}} dx \end{aligned} \quad (28)$$

Plugging back $P_s(l)$ from (21) and $\tau(l)$ from (28) in (19), we can thus estimate the expected delay for source node X located at a distance of ρ from the sink.

3.4 *Simulation Results for Uniform Random Deployment*

In order to validate the analysis in the previous section, we have simulated a network consisting of 700 nodes in MATLAB deployed according to uniform random distribution across a circular region of unit radius. We have obtained the plots of 1. Latency as a function of the duty cycle and 2. Latency as a function of the input data rate (event intensity/frequency). Events were generated using Poisson random process simulation, and each event generates a fixed packet size of 128 bits. Further, as in the previous case, independent duty cycling of nodes was employed, and various duty cycles were simulated (10% to 90%), for a Poisson event intensity of 15 packets per second. Further, the simulation was repeated with Poisson intensity of the events was stepped from 10 packets per second to 25 packets per second, with a fixed duty cycle of 33%. In both the cases, one can observe that the simulation results tallies with the analysis. Figure 20 shows a plot of the average latency for a node at a distance of 0.5 from the sink, as a function of the duty cycle for the given event intensity. One

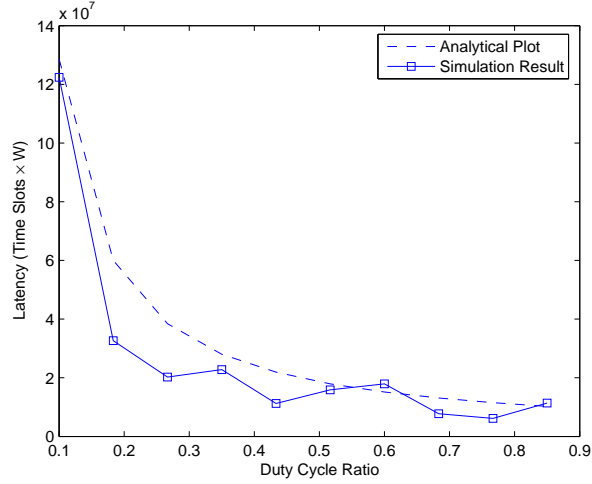


Figure 20: Simulation Results for Uniform Random Deployment - 1

can observe that as the event intensity reaches up to the capacity of the network, the average latency rises exponentially. We have also plotted the average latency for the same node vs the event intensity for a given duty cycle, in Fig. 21. Again, one can observe, as in the case of uniform grid deployment, that we incur a huge loss in latency for a relatively small gain in the energy efficiency, the gain in the energy efficiency arising as a result of lower duty cycle, resulting in savings on the idle energy wastage.

3.5 *Throughput Capacity of Sensor Networks*

In the context of ad-hoc networks, the transport capacity is defined as the of the average bits/sec that any node can transmit to a randomly chosen node in the network. Authors in [1] have obtained analytical bounds on network throughput and transport capacity for ad-hoc networks. It has been shown that for wireless networks with n nodes, where each node can transmit W bits per second in the channel, the throughput $\lambda(n)$ per node is given by $\lambda(n) = \frac{W}{\sqrt{n \log n}}$. However, in the scenario of sensor networks, we require a different definition for throughput capacity. This is because, the communication is not between a random pair of nodes, but from the event to the

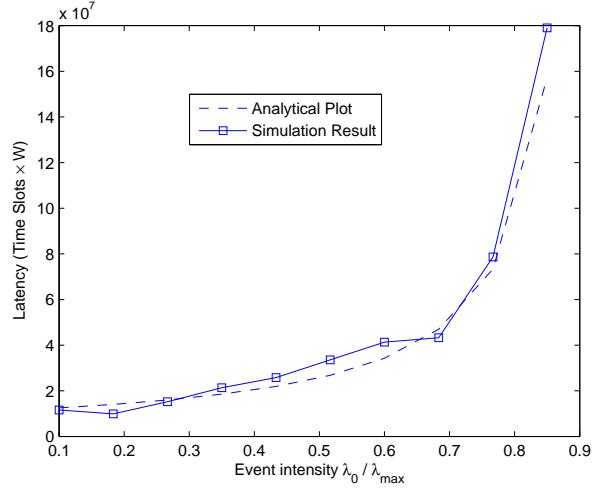


Figure 21: Simulation Results for Uniform Random Deployment - 2

sink, in a *convergecast* fashion. From our results in the previous section, we have a natural way of meaningfully defining the capacity of such networks - “the maximum intensity or frequency of event occurrence, as in bits/sec/m², that can be successfully sensed and transmitted to the sink in finite time”. This serves as a measure of the frequency and intensity of events that can be successfully handled by a sensor network. We expect that the throughput will be poorer for sensor networks than in general ad-hoc networks due to the aforesaid communication constraints. From the previous section, one can see that the sufficient and necessary condition for event dissemination to the sink in finite time can be obtained by making

$$\max \lambda(x) < \mu' \quad (29)$$

$$\Rightarrow \lambda(r(n)) < \frac{2W\pi}{n^2 r^2(n) BA} (2P_a - P_a^2) \times \frac{1}{3} n r^2(n) \quad (30)$$

$$= \frac{2W}{3B} \frac{1}{n r^2(n)} \frac{1}{\{(1 + \Delta)^2 + \frac{2}{\pi} (1 + \Delta)\}} \quad (31)$$

$$\Rightarrow \lambda_0 < \frac{2W}{3\pi B} \frac{1}{\{(1 + \Delta)^2 + \frac{2}{\pi} (1 + \Delta)\}} \quad (32)$$

Hence, the above bound gives the maximum event intensity (λ_0), within which all events can be reported to the sink within finite time. In time constrained networks,

one can also extend the above definition of network throughput capacity to “the maximum event intensity at which all event information is delivered to the sink with an average reporting delay of less than T_0 ”. An expression for the same can again be obtained from the latency expression derived in the previous section, but the analysis would be much more tedious. In any case, with the expressions from Section 3.3, one can obtain estimates of the time-constrained capacity using numerical computations.

3.6 Summary

In this work, we have derived closed form expressions for the average latency in distributed wireless sensor networks employing converge-cast communication paradigm. We have derived bounds and estimates for the latency for the grid deployment case, in terms of network parameters such as number of nodes, input data rate, channel bandwidth, duty cycle ratio and so on. We have shown extensions to estimate the same in networks with uniform random deployment of sensor nodes, using graph theoretical and probabilistic techniques. Our analytical bounds and estimates match well with the simulation results.

**MODELING AND ANALYSIS OF THE PERFORMANCE
OF COLLABORATIVE WIRELESS AD-HOC
NETWORKS: AN INFORMATION-THEORETIC
PERSPECTIVE**

VOLUME II

by

Ramanan Subramanian

CHAPTER IV

PRELIMINARY APPROACHES FOR MARKOV-CHAIN ANALYSIS OF SPMANETS

Delay tolerant networks (DTNs), forming a specific class of Sparse Mobile Ad-Hoc Networks (SPMANETs) are characterized by intermittent connectivity between nodes due to sparse node density and mobility, and, hence, long message delivery times [44]. Unlike traditional ad-hoc networks, any snapshot of such networks is almost always disconnected. In these networks communication and information exchange is possible between two nodes only when they are in proximity, in which case a *contact* is said to have occurred between them. Therefore, these networks employ communication paradigms relying on node mobility to carry data eventually to a destination in a time-bound fashion. Potential applications of DTNs include mobile sensor networks, disaster relief, military surveillance, and environment monitoring.

Intermittent connectivity, lack of end-to-end feedback, node mobility, and opportunistic communication in such networks make the problem of reliable and efficient message delivery in such networks very challenging. Conventional approaches to this problem involve the design of efficient routing schemes. While certain routing schemes designed for ad hoc networks such as DSR [29] and DSDV [45] fail in such networks, several efficient schemes tailored to these networks were devised recently [30–32, 46–48]. These schemes use the *store, carry, and forward* paradigm for message delivery. Certain other approaches use partial or complete knowledge of the network dynamics to effect efficient routing [30, 46, 49]. Message-ferry based schemes have been suggested in [31, 50, 51] where some mobile nodes that move in a predictable fashion, known as ferries, provide communication service in a deployment region.

Epidemic routing, aimed at minimizing latency, has been suggested as a viable solution to the problem of message delivery in DTNs [52, 53]. Here, multiple *identical* copies of messages are injected into the network, and node mobility is relied upon to transfer all the requisite data packets to the destination with high probability. Here, an intermediate node (other than the source and destination, called a *relay* node) transfers a copy of its packets to a node that it is in contact with if the latter does not already possess a copy of the same, resulting in both nodes possessing a copy of the same packet. Recent works have shown that a hybrid scheme that uses both epidemic routing and simple erasure codes achieves better performance [47, 54]. It was shown in [47] that such hybrid routing strategies employing both coding and replication are more robust than simple replication. For example, in [47], the authors consider schemes where data is first encoded with a replication factor of r and then packetized into sr chunks for some integer s . They are then relayed in a two-hop fashion. Simulations in [47] reveal the superiority of their scheme over other schemes such as *simple replication*.

The primary focus of this chapter is to provide ideas and the first steps towards evolving a methodology for general performance analysis of SPMANETs (also known as Intermittently-Connected Mobile Ad-Hoc Networks, or ICMANETs) under various communication scenarios, incorporating practical considerations such as contention in node-to-node communication, random contact durations, finite bandwidth, and finite buffer spaces in relay nodes.

There has been a lot of emphasis on computer simulations and algorithmic approaches to investigate the performance of SPMANETs and DTNs. In some cases, researchers resort to simulation tools for studying these networks. However, simulations are inadequate for a comprehensive understanding of the interplay of network parameters and scaling laws in these networks. Moreover, simulations may be very cumbersome for large network sizes and for some parameter choices. Another key

limitation is that simulations need to be repeated across several mobility models and communication protocols to fully understand the performance tradeoffs. Very little insight on the optimality of a given protocol can be gained through simulations. Hence, a framework predicting the performance of protocols under different mobility characteristics is needed.

Providing the first steps towards the above goals, this chapter specifically analyzes the *achievable throughput* of SPMANETs as an important performance measure. We consider the case where node mobility can be modeled as a random-walk process in a finite deployment region. This results in a *memoryless mobility* wherein future states of the nodes (position and direction of motion) do not depend on the past. The relay nodes are restricted by relatively small communication ranges, and by the finiteness of their buffers. We develop a Markov-chain based framework for the analysis of SPMANETs under single-copy unicast operation. We then apply novel state-reduction techniques on this chain to derive tractable chains from the original. This enables us to derive closed-form expressions for the throughput and the delay. The routing protocols used in the model fall under two categories: constrained two-hop relaying and unconstrained multi-hop relaying.

Our analysis shows that the traditional approach of modeling contacts for each pair as independent Poisson processes in SPMANETs deviates significantly from actual performance especially in the finite-buffer case. Hence, we establish the need for sophisticated modeling of performance modeling in SPMANETs which makes use of the knowledge of mobility characteristics beyond contact distributions and is tractable at the same time. Analytical results show that the effect on throughput due to finite buffers and sparseness compound each other, resulting in reduced throughput in the two-hop case. We finally validate our results using extensive simulations. Even though our initial models are simplistic, we find that a straightforward approach is often highly cumbersome. This points to the need for more sophisticated approaches,

which will be our goal in the succeeding chapters.

Performance analysis of Delay Tolerant- and Intermittently Connected Networks has been visited in the past by Spyropoulos *et al* and Jindal *et al* in [55,56]. However, they devise analytical models only as a means to prove the superior performance of routing protocols proposed by them. As a result, some of the issues that have been a motivating factor in this work are not carefully considered. For example, the authors in [55] compute the average delivery delay in the following way: Given that nodes are randomly walking on a 2D grid and that the node buffers are initially empty, we inject some fixed number of packets into the network, designating a few nodes as source nodes; what is the expected delay for these messages to reach the destination? On the other hand, we concern ourselves with the following question: Given that nodes are in steady-state mobility *and* the node buffers reach their corresponding steady states, what is the achievable throughput (and delay), given that packets have been generated continuously in the network by certain source and destination nodes? The two questions lead to very different performance measures, and the latter is the more frequently visited version of the problem in network information theory. In other contexts, the performance of DTNs have been modeled using Poisson-process approximations [57–59]. In these papers, definitions of network performance in DTNs fairly close to our own were introduced to study these networks in an information-theoretic manner. However, a major drawback of these works is Poisson-process based modeling, which affects their applicability severely. For a detailed discussion of the severe limitations in Poisson-process modeling, the reader is referred to the Motivation section (Section 4.1). In addition, the effect of finite buffer has not been studied in the past in the theory of SPMANETs, to the best of our knowledge. The results pertaining to this chapter are also available in our paper [60]¹.

¹This work was done in collaboration with my colleague Badri N. Vellambi, at Prof. Fekri's group.

4.1 Motivation

4.1.1 Finite-Buffer Considerations in SPMANETs

One of the key considerations in the design of practical SPMANETs involves the availability of storage-space at the nodes (devices). Existing literature in the field of SPMANETs identifies limited storage capacity as a critical design consideration [61–63]. To summarize from the literature, we make a case that finite-buffer considerations are critical due to the following reasons: (1) A node (device) may run several applications simultaneously wherein processes are allocated a certain limited chunk of resource by the operating, making storage a limited resource. (2) Even as storage capacities in physical devices are rising, communication bandwidths too are rising. Hence, finite-buffer effects on performance would still be felt. (3) Several key applications involving SPMANETs such as Delay/Fault-Tolerant Sensor Networks typically employ devices with much lower storage capabilities. For example, the MICA2 sensing device and the vehicular on-board unit OBU3021 shown in Table 2, which may potentially be used in VANET applications, are heavily constrained by physical memory size. Even in the case of high-end devices with high *physical* memory, the amount of *virtual* memory available for various services would be low, due to the first two reasons. Though the effect of finite memory is significant, an analytical study on its implications in SPMANETs is particularly lacking in the literature. Hence, we incorporate the provision for finite-buffer effects into our analytical framework.

Table 2: Storage capacities of some mobile devices

| Device name | Available memory | Transmission bandwidth | Reference |
|----------------|------------------|------------------------|------------------|
| MICA2 mote | 512 KB flash | 38.4 Kbaud | Wu et. al., [64] |
| Kapsch OBU3021 | 1200 bytes | 100kbps | [65] |

4.2 Limitations of Poisson-process Modeling

In this section, we shall explain why approximations to the modeling of SPManETs such as Poisson-process contacts that are commonly employed in some contexts [57–59, 66] do not suffice, leading to the need for a more sophisticated methodology for analysis. A critical issue in performance analysis of finite-buffer SPManETs is that the models and assumptions employed need to be able to track buffer variations and queuing effects in a fairly accurate manner. The authors in [57] argue that an accurate analysis of SPManETs can be obtained by such modeling. They validate this claim for the case wherein multiple copies of the same message are disseminated across the network to decrease message delay. Further, there is no restriction on the buffer size in their models. We show however that, in the case of limited node buffers Poisson-process models grossly overestimate the throughput.

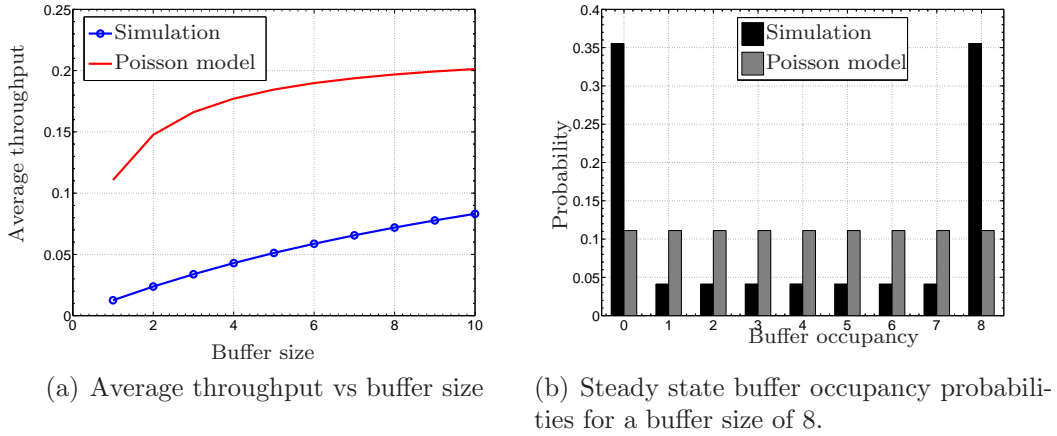


Figure 22: Comparison of Poisson method and exact analysis

An illustration of the inadequacy of Poisson-process mobility models is shown in Figs. 22(a), and 22(b). Here, we simulated a network deployed on a 20×20 square grid. One hundred mobile nodes were deployed in addition to the source and destination nodes, each of which performed an independent and identical random walk on the hypothetical grid, resulting in a random-walk based mobility model. We employed a two-hop single-copy routing for this network, wherein the source node transmits

a single block of packets to each relay node, and no such block is repeated. The throughput achieved (over a long period of time) at the destination under this scheme is plotted as a function of the buffer size in Fig. 22(a), for both the exact random walk simulation as well as for the equivalent Poisson model described in [57]. We see that the Poisson approximation deviates considerably from the exact simulation, by as much 100 %. The reason for this is clearly illustrated in a plot of the steady-state buffer occupancy probabilities (Fig. 22(b)). Here, the same parameters were used, and the buffer size was fixed as 8 blocks of packets. One can see that the Poisson approximation method fails to capture a critical that the simulation indicates: The network spends considerably longer periods in the empty-buffer state when it is in contact with the destination. As a result, the Poisson-based approximation heavily overestimates the throughput in finite-buffer conditions.

A rather interesting observation from the above discussion is that in general SP-MANETs, inter-contact times do not follow any known well-behaved distribution. This would mean that neither the packet reception process nor the packet transmission process are Poisson processes, since inter-arrival times for Poisson processes are necessarily exponentially distributed (in the continuous time case) or geometrically distributed (in the discrete-time case). Accurate models for the distribution of contact times in such mobility scenarios are not available to the best of our knowledge. We show that even with such unknown contact time distributions, which is the case in SPMANETs with certain mobility patterns, exact analysis of network performance is quite possible. On the other hand, exact analysis in much of queuing theory deals with systems wherein at least one of the processes can be modeled as Poisson arrivals [67]. Due to the aforementioned reasons, these results clearly will not apply to even SPMANETs with simple memoryless random walk mobility models. Instead of applying approximations from general queuing theory to our case, we show that model-specific analysis can sometimes yield exact results.

4.3 Models and Definitions

We first explain in brief some of the terminology in Markov chains adopted in this work, since some of may not be fairly standard, for the convenience of the reader. A finite-state Markov chain is *periodic* if for some state i , any return to that state must occur in multiples of k time-steps for some $k > 1$. A Markov-chain is said to be *irreducible* if for no two subsets S_1, S_2 of states, it is impossible to reach any state in S_2 starting from some state in S_1 . A finite-state Markov chain is said to be *ergodic* if and only if it is both irreducible and aperiodic. In our context, we will assume a random walk mobility model whose Markov chain representation is ergodic. A Markov chain $X(t)$ is said to be *reversible* if $Pr[X(\tau + 1) = j | X(\tau) = i] = Pr[X(\tau) = i | X(\tau + 1) = j]$, where i and j are any two specific states in the chain.

4.3.1 Network and node mobility model

We model the SPMANET as a deployment consisting of a given number of nodes performing *natural random walk* on a uniform two-dimensional grid in discrete time. We note that this is a space-time discretized version of the Brownian motion model defined in a finite region. The mobility characteristics of nodes in the network are assumed to be independent and identical to each other in nature. By the term *natural random walk*, we mean that a mobile node at any time can choose to remain in the same location in the grid, or choose to move to an adjacent location in the next time step.

Throughout the chapter, we deal with the *reversible uniform grid mobility model* as a special case of mobility. Here, the deployment region consists of an $M \times M$ square grid on a 2-D plane. The random walk is in discrete time, with a constant step size of T . There is an assignment of transition probabilities such that the resulting Markov chain is both *reversible* and *uniform*, as shown in Fig. 23(b). Though not indicated, the transition probabilities for i to j is the same as from j to i for any two points

in the grid i and j . We use this assignment of transition probabilities (weights) for convenience in analysis. Slight changes in the assignment of transition probabilities will only have marginal effects on the results. Clearly, the resulting Markov chain for the random walk model is also ergodic. Equivalently, we can also consider the deployment region to be a unit square region divided into equal M^2 square cells, as shown in Fig. 23(a), wherein nodes walk between adjacent cells, rather than vertices. We shall adopt this interpretation for the rest of this section.

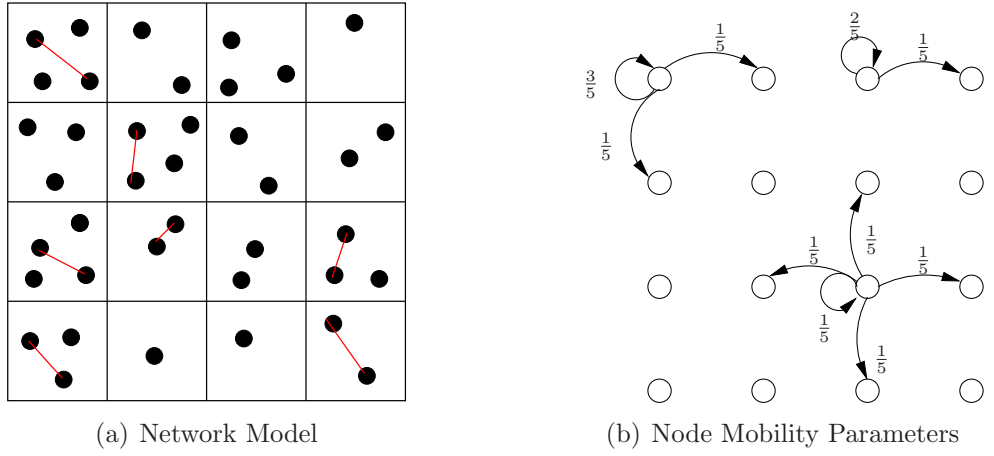


Figure 23: Mobility Model

Two nodes will be able to communicate with each other only if they are in the same cell. Further, in order to ensure collision-free communication, we assume that when two nodes in a cell communicate, the MAC protocol makes sure that all other nodes are silent. A communication link lasts for the period T , same as the step size in the discrete random walk. We employ a simple link model for the preliminary discussion. Each communication link leads to successful transmission of 1 packet between nodes, i.e. in our model, the wireless channel is lossless and time-invariant. We further assume that the buffer sizes of the nodes are limited to B such packets.

4.3.2 Packet transfer protocols

Next, we describe the packet routing protocols used in our analytic modeling. We assume that messages at the source are packetized, and that the source tries to transmit an unlimited number of them continuously to the destination. Throughout this work, we assume that the source node never transmits multiple copies of the same packet. Hence, at any given time, there is at most one copy of the packet in the network. In the past chapter, we have shown that single-copy routing can indeed be preferable to multi-copy routing especially in scenarios where messages can be sent by using erasure codes, in the context of Delay-Tolerant Networks. A contact is said to be “successful” if the node-pair in contact win the contention phase and thus get to communicate with each other. A mobile (relay) node, after accepting packets from the source, retains the same until a successful contact with a destination node occurs. Contacts that may occur between two mobile (relay) nodes elsewhere are ignored. It is further noted that in this scheme, the node buffers use FIFO policy, i.e., packets are transferred to other nodes in the order they were received.

4.3.3 Contention resolution

The contention resolution policy is described as follows:

- At any given location in the network, at most one pair of nodes is allowed to communicate. All other co-located nodes remain silent.
- Whenever the source and destination are in the same location, the source successfully transfers a single packet to the destination node.
- If several relay nodes meet a source node and the destination node is not in the same location, one of the relays is selected at random and the source node tries to transfer one packet to the relay.
- If several relay nodes meet a destination node and the source node is not in the

same location, one of the relays is chosen at random and it transfers one packet from its buffer to the destination, if possible.

4.4 *Description of the General Approach*

In this work, we are interested in the quantity defined as the *throughput* of the network: We start the network at some particular state. We employ a particular communication paradigm, and packets are transmitted across the network. After a sufficiently long time (after which we assume that the network is in its steady state) during which communication happens continuously, we observe the network for a duration of τ . We are interested in the following questions:

Throughput capacity: For a given source-destination pair, say s and d , the expected rate at which the network transfers packets from s to d . In other words if $\mathcal{N}_{s,d}(\tau)$ packets are transmitted from s to d in time τ , the throughput capacity is:

$$\mathcal{C}_{s,d} = \frac{\mathcal{N}_{s,d}(\tau)}{\tau}$$

Average latency: The expected amount of time a packet has to spend in the relays nodes' buffers before reaching the destination.

We can construct the *state* of the network as follows. Assume that the network has n nodes performing a random walk defined on a weighted graph G independently. Let $X(t) = (X_1(t), X_2(t), X_3(t), \dots, X_n(t))$ be the locations of the nodes at times $t > 0$ and let $\mathcal{M}(t) = (\mathcal{M}_1(t), \mathcal{M}_2(t), \mathcal{M}_3(t), \dots, \mathcal{M}_n(t))$ be the buffer occupancies of the nodes at the end of all communication in that time slot. Then the pair $(X(t), \mathcal{M}(t))$ uniquely identifies the state of the network at time t . Clearly, given $(X(t), \mathcal{M}(t))$, we know that the probability of transition to $(X(t+1), \mathcal{M}(t+1))$ is defined from the

mobility model and the protocol model. This results in a Markov chain which uses full knowledge of the network state. The above network-level Markov chain will be ergodic if the random walk itself is ergodic, which is the case in our work. Let Φ be the state transition matrix for the chain tracking $(X(t), \mathcal{M}(t))$, and let ψ be its steady state distribution. A point to note here is that the position-marginal distributions ψ_{X_j} pertaining to the positions of nodes j will be independent and identical to the random walk steady state statistics. However, in general, the buffer-marginal distributions $\psi_{\mathcal{M}_j}$, though identical, will neither be independent of each other, nor be independent of the node locations.

There are states in the network-level chain wherein packets are delivered to the destination node(s); these are designated as *desirable states* for our purpose. To derive the throughput capacity of this network, it is thus sufficient to obtain the steady state distribution ψ :

Theorem 1 *Let q_j be a desirable state for a given source destination pair (s, d) , and let $\nu(q_j)$ be the number of packets delivered to d that originated from s in each of the states. Then, the throughput of the network is given by:*

$$\mathcal{C}_{s,d} = \sum_j \nu(q_j) \psi_{q_j}$$

Proof: This follows directly from ergodicity, due to which the frequency of visits to q_j is ψ_{q_j} . Hence, the number of packets delivered in an asymptotically long time τ is $\sum_j \nu(q_j) \psi_{q_j} \tau$.

Since the state space representation for the network is generally huge, it is very difficult in general to compute the steady state distribution through direct analysis. We tackle this situation by using reduction methods and estimation techniques on this chain, as we show in the following sections.

4.5 *Two-hop single-unicast with immobile source and destination*

Our first case of analysis involves a pair of immobile source and destination nodes that are fixed at two diagonal corners, i.e., $(0, 0)$ and $(M - 1, M - 1)$ respectively in a $M \times M$ grid network. We label this case as ISD_2 , meaning Two-hop with Immobile Source and Destination node pair. One can also repeat the same analysis for other source/destination locations with virtually no difference in the methodology. Let us say n mobile relay nodes are deployed in the network, each with a buffer limit of B blocks. We are interested in deriving the throughput capacity of this network. For our initial discussion, we ignore the effect of contention and will introduce it later. Effectively, this is a hypothetical situation wherein the source (destination) is capable of transmitting (receiving) one block to (from) each of the node it is in contact with, within the same time slot.

Relay nodes pick up packets on meeting the source and drop it off on meeting the destination, acting as relays. A relay is not allowed to transmit its packet to another relay, resulting in a two-hop restriction. As stated in the previous section, we construct the network-level Markov chain to enable us in the analysis of throughput.

An important observation for this two-hop network is that it enables us to derive the throughput on a per-node basis independently of the other nodes. Further, as the network is homogeneous, the throughput contributions are equally distributed among all nodes. This results in a three-dimensional Markov chain is depicted in Fig. 24, and clearly has $M^2(B + 1)$ states. Each layer represents a different occupancy state b for the buffer. Further, the node locations (except the source and destination locations) in the grid are to be replicated at each layer, as shown. The transition probabilities are not indicated, it suffices to say that they are exactly the same as the corresponding transition probabilities for the random walk.

A little clarification on the nomenclature used for states follows here. In Fig. 24,

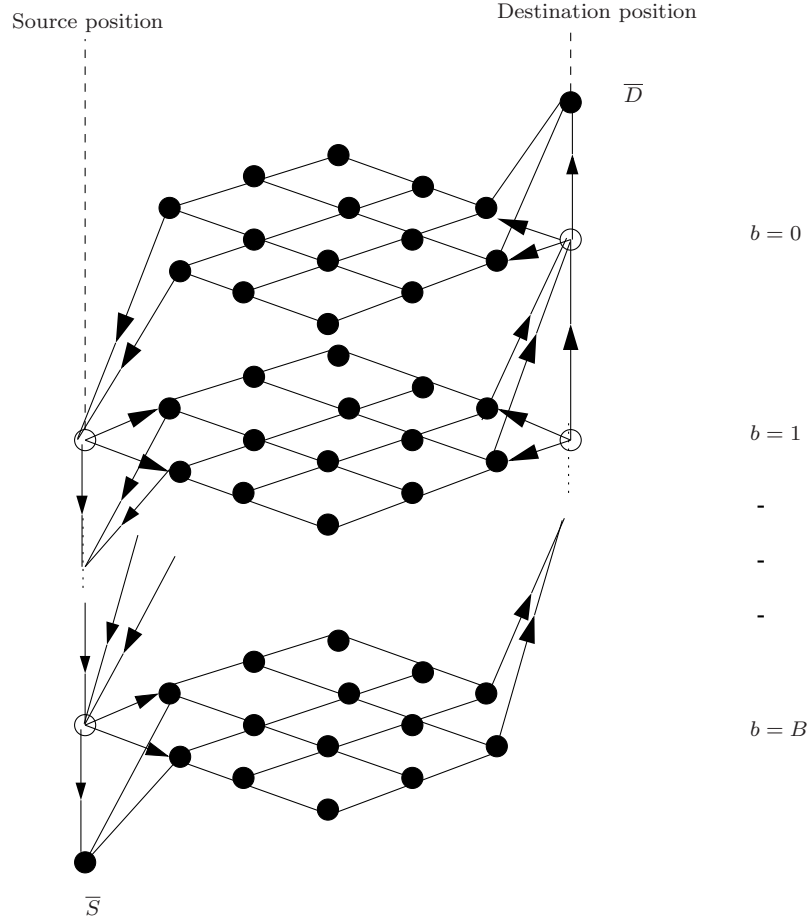


Figure 24: Markov chain for the two-hop grid network (ISD_2)

states indicated by a solid bubble have self-loops, and those indicated by hollow bubbles do not. The hollow bubble states are designated as *active states*, wherein the node picks up/delivers packets from the source/destination. These states are designated as S_1, S_2, \dots, S_B and D_0, D_1, \dots, D_{B-1} . There is no state designated as S_0 or D_B since a relay node cannot have zero/full buffer after successful contact with the source/destination node. States \bar{S} and \bar{D} are *non-contributing states*, corresponding to saturated and empty buffer state while the node meets the source and destination respectively. All other states in the ISD_2 chain are *passive states*, and are designated by the two-tuple (x, b) corresponding to the location of the node in the grid and the buffer occupancy. Detailed analysis and results for the two-hop immobile source-destination case(ISD_2) are presented below.

Theorem 2 *The throughput of the ISD_2 network with node buffer sizes B , grid size M^2 , and n mobile nodes is given by:*

$$\mathcal{C}_{s,d} = \frac{n}{M^2} \left(\frac{B}{B + \left\lceil \frac{5\gamma(M)+8}{2} \right\rceil} \right) \approx \frac{n}{M^2} \left(\frac{B}{B + 6 \log M} \right)$$

where $\gamma(M) = 4 \sum_{k=0}^{M^2-2} \varphi(k, 0)$, $\varphi(k, l)$ being the solution to the two-dimensional difference equation system

$$\begin{aligned} 4\varphi(k, l) &= \varphi(k-1, l) + \varphi(k+1, l) \\ &+ \varphi(k, l-1) + \varphi(k, l+1) \end{aligned} \quad (33)$$

$$\forall 0 \leq k, l \leq M^2 - 3 \quad (34)$$

$$\varphi(M^2 - 3, k) = 0 \quad \forall k : 0 \leq k \leq M^2 - 1 \quad (35)$$

$$\varphi(0, 0) = \varphi(M^2 - 4, M^2 - 3) = \frac{1}{2} \quad (36)$$

The above theorem is proved using various steps, by means of the following Lemmas:

Lemma 1 *The stationary probabilities for the active states in the ISD_2 Markov chain are equal i.e., $\psi_{S_k} = \psi_{D_l} := V_0 \quad \forall 1 \leq k \leq B, 0 \leq l \leq B - 1$.*

Proof: Let us assume that $\psi_{S_1} = V_0$. By symmetry, $\psi_{D_{B-1}} = V_0$ by symmetry, since the distributions of contact times with the source and destination are equal. The steady-state distribution of the remaining $M^2(B-1) + 2$ states in the layers $b = 1, b = 2, \dots, b = B-1$ will only depend on each other and ψ_{S_1} and $\psi_{D_{B-1}}$. With very little mathematical manipulation, one can show that the steady state probabilities of these remaining states is exactly the average of all the *neighboring* states. We will then have $M^2(B-1) + 2$ steady state equations with as many unknowns, and the solution to this should be unique, since these are simultaneous linear equations. It remains to show that the solution V_0 for each one of these states is valid, which it is due to the neighbor-average property of the steady state probabilities, or it can be verified easily by inspection.

Corollary 1 *The steady-state distribution for passive states (x, b) in the ISD_2 chain, where $1 \leq b \leq B-1$ are equal to the active states' steady state probability.*

Proof: Follows from the proof of Lemma 1.

Thus, we have fixed the steady-state distribution for $M^2(B-1)$ out of the $M^2(B+1)$ states that we need to determine. Hence, we only need to determine the steady state distributions for the remaining $2M^2$ states belonging to layers $b = 0$ and $b = B$. Further, we have the following relation:

Lemma 2 *For the ISD_2 Markov chain, the steady-state distributions satisfy the following property:*

$$\forall x \notin \{(0, 0), (M-1, M-1)\},$$

$$\sum_{b=0}^B \psi_{(x,b)} = \psi_{\bar{S}} + \sum_{k=1}^B \psi_{S_k} = \psi_{\bar{D}} + \sum_{l=0}^{B-1} \psi_{D_l} = \frac{1}{M^2}$$

Proof: This follows from the fact that each of the above sums represent the *marginal distributions* of the steady state *w.r.t.* the location in the grid. Since the random walk is independent of the buffer states, these marginal probabilities are exactly the

same as the steady-state probability distribution for the random walk itself, each of which is $\frac{1}{M^2}$ in our model.

It now follows from the above Lemmas that we only need to calculate the probability distribution for the state \overline{D} in order to estimate the throughput. To do so, we will use the so-called *potential method* on the $b = 0$ layer to obtain the steady-state distributions of the remaining states in the ISD_2 chain, having known that the probabilities for states S_1 and D_0 are V_0 each. Knowing this, one can easily see that $\psi_{(0,1)} = \psi_{(1,0)} = \frac{5V_0}{2}$. Due to the well-known connection between reversible Markov chains and electrical networks [68], the problem boils down to the computation of potentials in an electric mesh formed by a grid of unit resistances, as shown in Fig. 25. We just need to calculate the potential at D_0 in the figure. Further the resistive mesh itself can be reduced to its equivalent $g(M)$ (Fig. 25). The corresponding steady-state equations then reduce to:

$$\left(V_Y - \frac{5V_0}{2}\right)g(M) + (V_Y - \psi_{\overline{D}} - V_0) = 0 \quad (37)$$

$$\psi_{\overline{D}} = \frac{3V_0 + 2V_Y}{2} \quad (38)$$

for which the solution is:

$$V_Y = 5\frac{1 + \gamma(M)}{2}V_0 \quad (39)$$

$$\psi_{\overline{D}} = \frac{5\gamma(M) + 8}{2}V_0 \quad (40)$$

In the above equations, $\gamma(M) = \frac{1}{g(N)}$. Exact computation of $\gamma(M)$ is beyond the scope of this work, and can be obtained using Laplacian methods described in [69], and the difference equations involved are summarized in Theorem 2. For a fairly large grid, 25×25 or more, it can be shown (either by exact analysis or using numerical tools) that $\left[\frac{5\gamma(M)+8}{2}\right] \approx 6 \log M$. Thus, we can compute the exact expression for the throughput due to a single node from Lemmas 1,2 and (40), completing the proof of

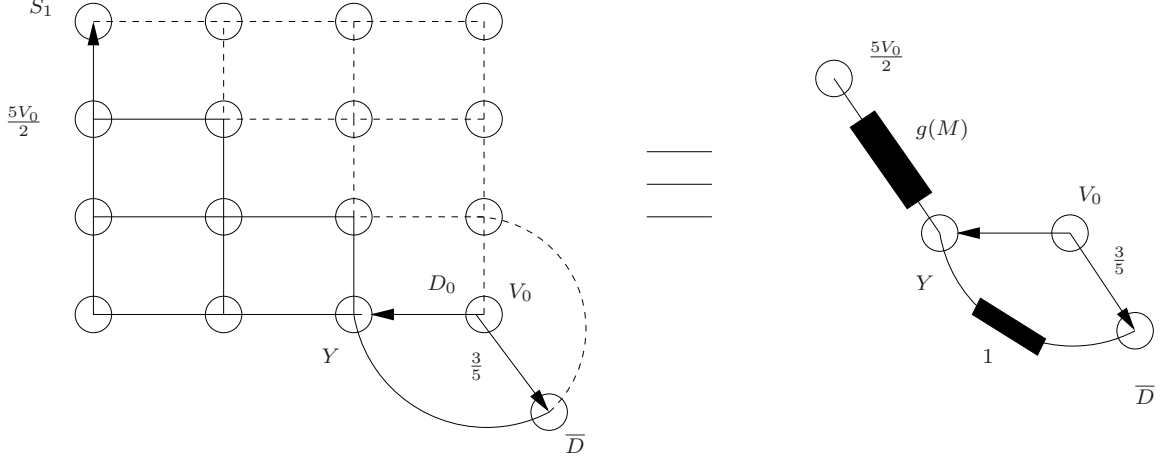


Figure 25: Solution of the ISD_2 chain by Potential method

Theorem 2 as follows:

$$\frac{\mathcal{C}_{s,d}}{n} = \sum_{l=0}^{B-1} \psi_{D_l} = \frac{1}{M^2} \frac{\sum_{l=0}^{B-1} \psi_{D_l}}{\psi_{\bar{D}} + \sum_{l=0}^{B-1} \psi_{D_l}} \quad (41)$$

Next, we incorporate the effect of contention into our model. In the resulting network, every time a given node meets the source or destination node, it encounters some contenders for the finite bandwidth resource. Using the ASTA theorem [70], one can show that the ISD_2 Markov chain can again be reduced to that from a single node's perspective. We will then have additional states in the new chain that are designated $\bar{S}_1, \bar{S}_2, \dots, \bar{S}_B$, and $\bar{D}_0, \bar{D}_1, \dots, \bar{D}_{B-1}$. A sample state addition for a part of the original chain is shown in Fig. 26. The additional states correspond to the event that a relay node loses contention due to the presence of competing nodes in the same cell location, leading to no change in its buffer state. The Markov chain for this network largely remains the same, with the exception of these additional states and corresponding modifications to the transition probabilities, as shown in the figure. Here p_c is the probability that a node wins contention on meeting the source (or) destination, which can be easily computed from the steady state random walk parameters. The analysis of the resulting chain is quite similar to the previous

contention-free case and is omitted to avoid redundancy. We then have the following result:

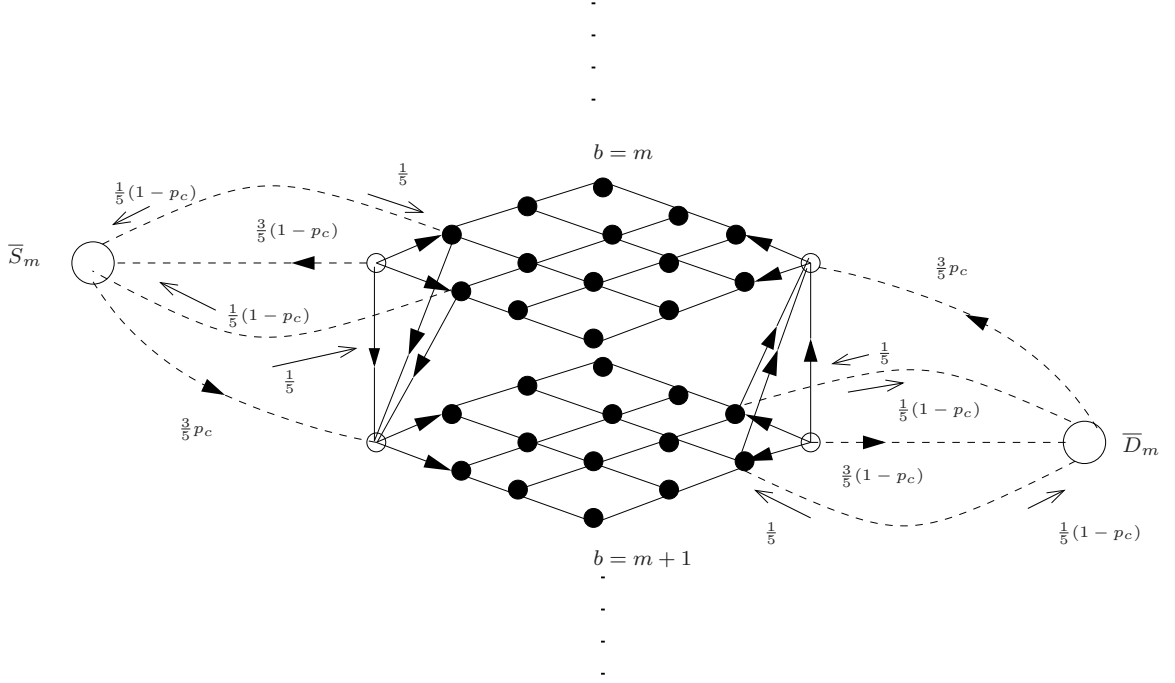


Figure 26: State-space modification to include contention

Theorem 3 For ISD_2 network restricted by contention, the total throughput is given by:

$$\mathcal{C}_{s,d} = \frac{np_c}{M^2} \left(\frac{B}{B + (1 - p_c) + p_c \left[\frac{5\gamma(M)+8}{2} \right]} \right)$$

where $p_c = \frac{M^2}{n} \left[1 - \left(1 - \frac{1}{M^2} \right)^n \right]$.

Here, $\gamma(M)$ is identical to the previous case.

We now derive the average packet delay of the network. It can be immediately derived by direct application of *Little's law* [67] to our network. We will then have the following expression for latency as a result:

$$\mathcal{L} = \frac{M^2}{2p_c} \left[B + (1 - p_c) + p_c \left\{ \frac{5\gamma(M) + 8}{2} \right\} \right]$$

An observation from the above analytic result is that the finite-buffer and sparseness together have a compounded throughput reduction effect in SPMANETs. Further, as in the case of general ad-hoc networks, the per-node throughput decreases with increase in n even though the total throughput in the network increases. However, interestingly, this decrease is accompanied by a slight mitigation of the limiting effect due to finite buffers.

4.6 Two-hop single-unicast with mobile source and destination

Since exact modeling of mobile-source mobile-destination unicast under the two-hop scheme involves construction of a Markov chain tracking both the location of the node, source, destination and the occupancy of the buffer, the number of states in one such Markov chain will be exactly M^6B states. Exact analysis of the mobile source-destination case will be dealt with later. As a simplification, consider an approximate Markov model MSD_2 for this network setting from the perspective of an relay node wherein each relay node keeps track of its buffer $\mathcal{M}(t)$ and an additional partial location information that is a quaternary-valued random variable $\mathcal{R}(t)$ defined as follows.

- $\mathcal{R}(t) = 1$ if at time t , the node shares the same location as the source.
- $\mathcal{R}(t) = 2$ if at time t , the node shares the same location as the destination.
- $\mathcal{R}(t) = 3$ if at time t , the node shares the same location as both the source and the destination.
- $\mathcal{R}(t) = 4$ if at time t , the node shares the same location with neither the source nor the destination.

The fundamental idea behind such a model is the fact that under the chosen random walk pattern, the buffer occupancy does not depend on the exact location but only

the instance of the random variable \mathcal{R} at that time. Under MSD_2 , we have to keep track of only $4(B + 1)$ states, a drastic reduction in the complexity of the original problem without an oversimplification of the tradeoffs pertaining to the problem, as we shall note in Section 4.7, where we present our results of simulation. Since we track neither the exact positions of the relay node nor that of the source or destination, we model the transition of the partial location information using an “average” model based on the steady-state probabilities of the nodes. This average model for tracking the variation of $\mathcal{R}(t)$ shall assume the nodes are distributed in the network locations with their respective steady-state probability. For instance, in order to compute $\Pr[(\mathcal{R}(t + 1) = 1)|(\mathcal{R}(t) = 1)]$, we may assume that the locations of both relay node and that of the source is given by their steady state distribution. Hence, the probability of this event can be calculated by averaging the probability that both the relay and the source nodes move from the same node to another or remain together at the location they were at time t . Equivalently,

$$\Pr[(\mathcal{R}(t+1)=1) \wedge (\mathcal{R}(t)=1)] = \sum_{l,l'=1}^{M^2} \frac{1}{M^2} P_{ll'}^2 \quad (42)$$

$$= \frac{5M^2 + 8M + 8}{25M^2}. \quad (43)$$

The computation for conditional probabilities other values can be performed similarly. Using these computed average marginal transition probabilities, packet transmission protocol for two-hop and contention resolution model as described in Section 4.3, we can model the various transition probabilities $\Phi((b', r')|(b, r))$ for various values b, b', r, r' in the Markov chain MSD_2 . The dynamics of this model that are obvious are enlisted below.

$$1. \quad \Phi((b', 1)|(b, r)) = 0 \text{ if } b' < b. \quad (44)$$

$$2. \quad \Phi((b', 2)|(b, r)) = 0 \text{ if } b' > b. \quad (45)$$

$$3. \quad \Phi((b', r')|(b, r)) = 0 \text{ if } r', r \in \{3, 4\} \text{ and } b \neq b'. \quad (46)$$

Once the exact Markov chain is constructed using the parameters, one can directly compute the steady-state probabilities for the states of the chain, and thereby the throughput of the two-hop scheme for a network of mobile source and destination nodes using the MSD_2 model in the following manner.

Theorem 4 *Consider the MSD_2 for a network of n relay nodes each with a buffer size of B in a grid with $M \times M$ vertices. Then, the throughput between the source node s and the destination node d is given by*

$$\mathcal{C}_{s,d} = \frac{np_c}{M^2} \frac{B}{B + 1 - p_c + \frac{1 - \beta(M) + \alpha(M)(1 - p_c)}{\beta(M)}} \quad (47)$$

where $\alpha(M), \beta(M)$ are parameters depending only on the random walk transition matrix P .

Proof: For the sake of clarity, we provide just the outline of the proof. Note that just as before the computation of throughput follows from the identification of the steady-state probabilities for the MSD_2 chain. So, let $\psi : \{0, 1, \dots, B\} \times \{0, 1, 2, 3\} \rightarrow [0, 1]$ to denote the steady-state probabilities of the MSD_2 chain. Since the state transition for \mathcal{R} uses the steady-state distribution probabilities for the locations, we can verify that

$$\Pr[\mathcal{R} = r] = \begin{cases} M^{-4}(M^2 - 1) & r = 1, 2 \\ M^{-4} & r = 3 \\ M^{-4}(M - 1)^2 & r = 4 \end{cases} \quad (48)$$

Also, using the properties of Φ and the (48), we can conclude the following.

$$\sum_b \psi(b, r) = \Pr[\mathcal{R} = r] \quad (49)$$

$$\psi(b, 3) = \frac{\psi(b, 1) + \psi(b, 2)}{2(M^2 - 1)} \quad (50)$$

$$\psi(b, 4) = \frac{(M^2 - 1)}{2} (\psi(b, 1) + \psi(b, 2)) \quad (51)$$

Note that (49) simply follows from the fact that the marginal for \mathcal{R} can be obtained by summing over all possible buffer occupancy possibilities. Also, while the scaling

factors for (50) and (51) are obtained from the marginal distribution, the actual relation follows from the structure of Φ and they can be intuitively understood thus. Since the distribution of a node does not vary for states that have $\mathcal{R} = 3, 4$, the distribution of the buffer distribution depend on the probability of arriving at these states from $\mathcal{R} = 1, 2$. However, the probabilities of arriving to states with $\mathcal{R} = 3, 4$ equiprobably from those with $\mathcal{R} = 1$ and $\mathcal{R} = 2$. Other properties that can be shown

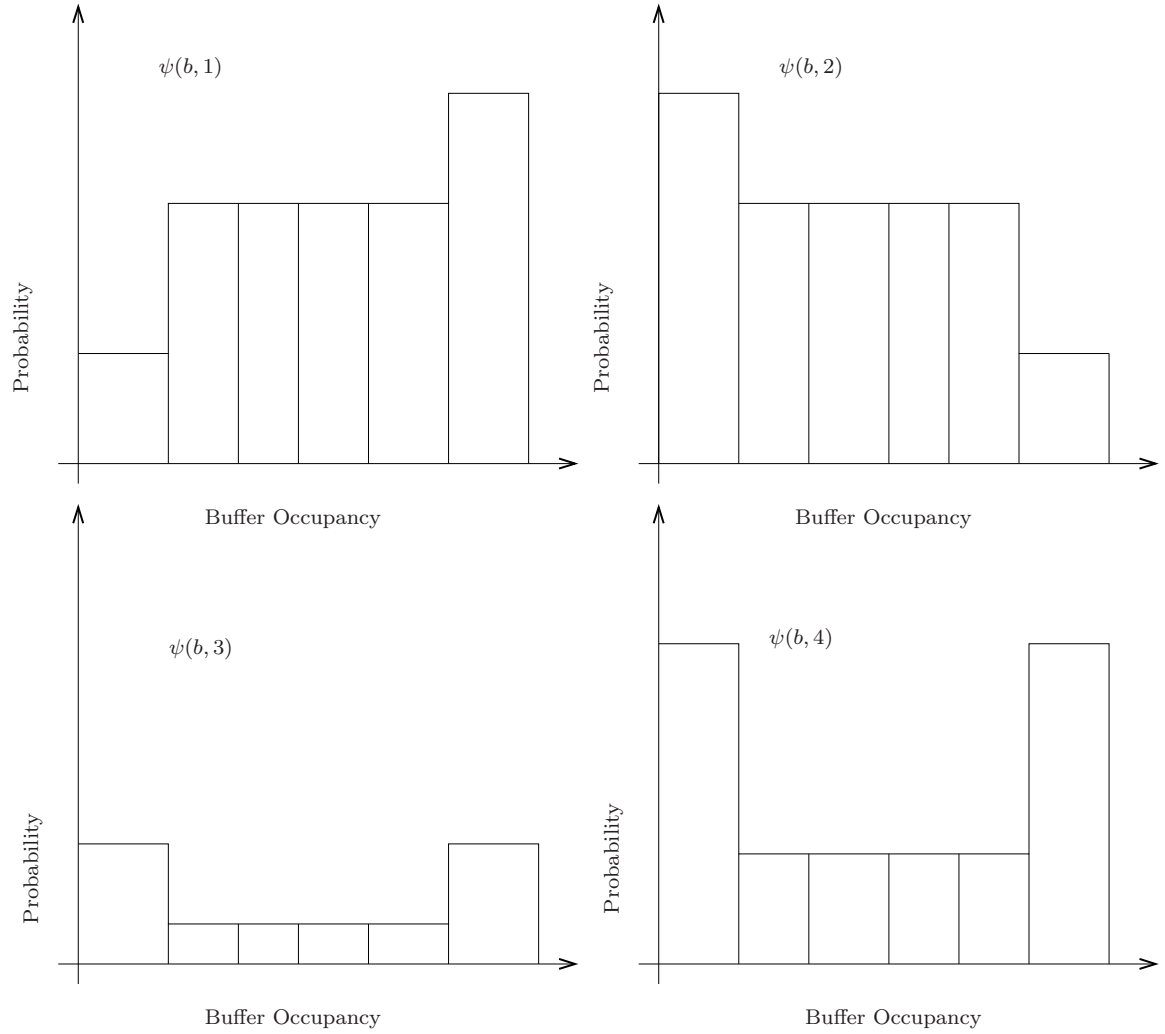


Figure 27: Illustration of the Steady-State Distribution Solution obtained for the two-hop mobile source mobile destination unicast problem.

exploiting the centrosymmetric structure of Φ but are not detailed here include the

following

$$\psi(b, 1) = \psi(B - b, 2) \quad (52)$$

$$\psi(1, 1) = (1 - p_c)\psi(2, 1) \quad (53)$$

$$\psi(1, 2) = \psi(1, 3) = \dots = \psi(1, B - 1) \quad (54)$$

$$\psi(1, B) = \frac{1 - \alpha(M)(1 - p_c)}{\beta(M)}\psi(1, 2) \quad (55)$$

Note that in (55), the constants $\alpha(M), \beta(M)$ are computed solely using the knowledge of the underlying probability transition matrix P pertaining to the random walk of the nodes on the network and are independent of the buffer occupancy parameter. The above properties suffice to uniquely identify $\phi(b, r)$ for all values of \mathcal{M} and \mathcal{R} . As an illustration, Fig. 27 presents the typical plot of the steady-state distribution as a function of the buffer occupancy and the partial location information \mathcal{R} . Finally, using the computed ψ , one can compute the throughput just as we did for the immobile source-destination situation.

4.7 *Simulation Results*

In our simulation set-up we studied three different cases of throughput variation common to both the schemes employed (ISD_2 and MSD_2). For the first case, we vary the buffer size of the node from 0 to 10, in a 20×20 network with 100 nodes. For the second case, the grid size is varied from 8×8 to a 30×30 , keeping the density of nodes ($\frac{n}{M^2}$) constant at 0.5, and for a fixed buffer size of 8 packets per node. Finally, the variation of throughput with the number of nodes(n) is studied for a fixed grid size of 20×20 and a fixed buffer size of 8. We discuss the results obtained for each of the schemes in detail below.

Plots of the average throughput from analysis and simulations for the ISD_2 (two-hop with immobile source-destination) case are shown in Figs. 28, 29, and 30. It is seen that the simulation results closely matches the analytic plots, confirming with

the claim of exactness of our the analysis. Further, the parameters in Fig. 28 are the same as those use for plotting Fig. 22(a), and both show the variation of throughput with buffer size. While the Poisson model failed, our model is very close to accuracy. Figure. 29 shows the variation of throughput with grid-size for a fixed density of nodes (here, density = number of nodes/number of grid vertices). The proposed Markov model clearly captures the effect of diminishing throughput; increasing node density needs to be considerably increased to compensate for sparseness, owing to contention effects. Fig. 30 shows the variation of the throughput with the node population n . The simulation confirms the fact that even though the total network throughput is increased, the per-node throughput actually decreases with increase in n due to contention effects.

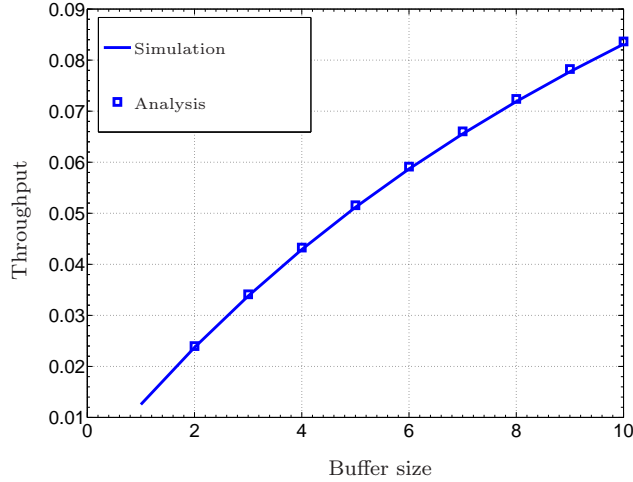


Figure 28: Throughput vs buffer size (fixed source/destination)

For the MSD_2 case (two-hop with mobile source/destination), the corresponding results for the same parameters as was used in the ISD_2 case are shown in Figs. 31,32,and 33. Here, the trends in general are similar to the ISD_2 case. However, the prediction error from analysis is slightly higher, as a result of the partial independence assumption introduced in the analysis in order to keep the state space

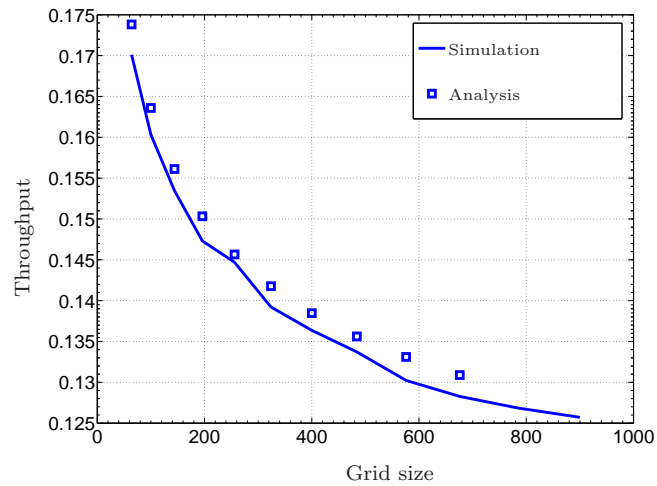


Figure 29: Throughput vs grid size (fixed source/destination)

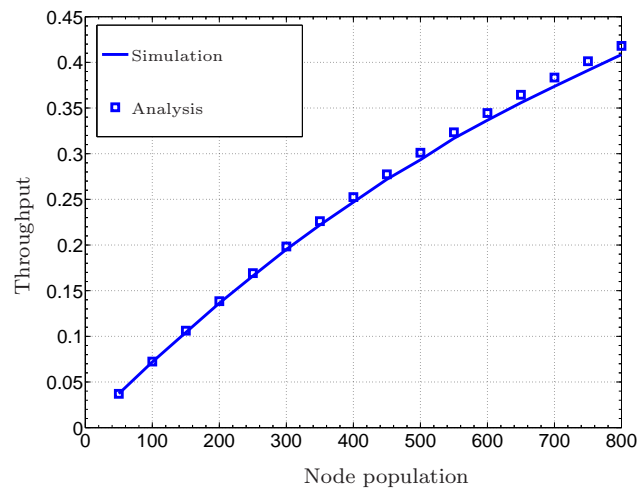


Figure 30: Throughput vs Node population (fixed source/destination)

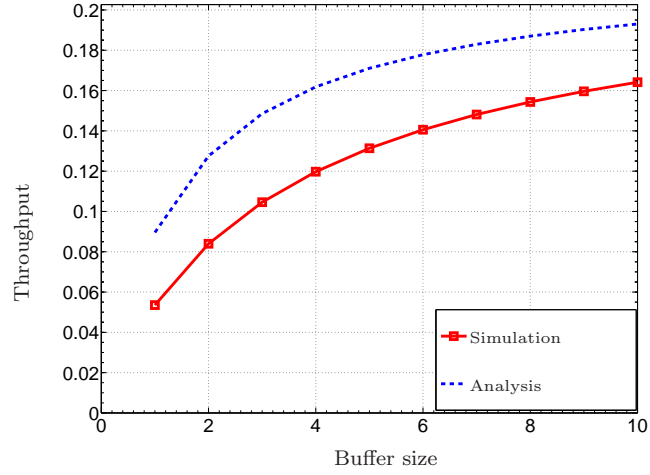


Figure 31: Throughput vs buffer size (mobile source/destination)

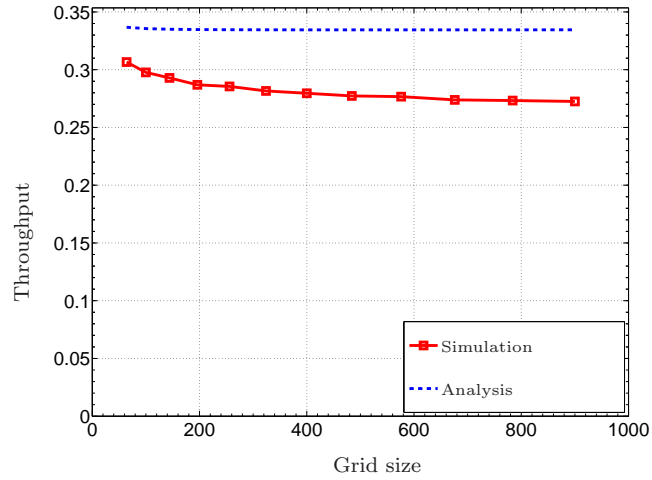


Figure 32: Throughput vs grid size (mobile source/destination)

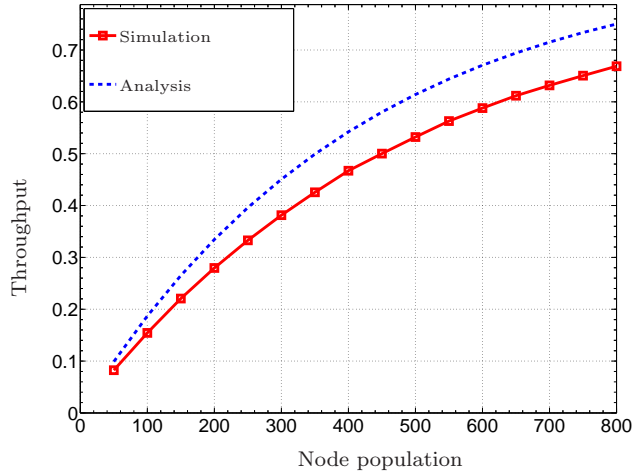


Figure 33: Throughput vs Node population (mobile source/destination)

tractable. In any case, the partial independence assumption, unlike the Poisson contact assumption, does track the throughput with a reasonable error margin (10 – 15% from the plots) for the cases of varying buffer and varying node population. However, the prediction shows virtually no variation with the grid size in Fig. 32. It must be noted that in this case, the node population density (i.e., $\frac{n}{M^2}$) is kept constant. The apparent lack of variation can be explained as follows. In the analysis of throughput for the MSD_2 case, the dependence of throughput on M for fixed density is solely captured by the ratio $\frac{\alpha(M)}{\beta(M)}$, which can be shown to approach 1.25 asymptotically. However, from the actual simulations, this dependence decays as $\Theta(\frac{1}{\log M})$, as in the fixed $s - d$ case. Still, the partial independence model for MSD_2 case is much better than the Poisson-contact approximation model, for which errors are high.

4.8 Summary

In this chapter, we presented a novel stochastic approach to the performance analysis of SPMANETs using Markov chain techniques. The throughput (and average delay) were derived for the case of two-hop unicast communication, incorporating finite buffer, finite bandwidth, and contention effects. The analysis involved new state-space

reduction and simplification techniques for Markov chains with large state spaces. For the two-hop case with immobile source and destination, the modeling matches actual simulations accurately, and for the case with mobile source and destination, fairly effective estimates were obtained. Further, this model outperforms Poisson-approximated contact methods. Especially in cases where the latter gives very poor estimates, our modeling gives estimates within reasonable error levels.

CHAPTER V

EVOLVING A GENERALIZED APPROACH FOR THE ANALYSIS OF SPMANETS

In the last chapter, we presented a simple SPMANET network model that involved a single immobile source-destination pair initially. The approach involved the construction of a Markov chain to describe the communication activity in the network, and whose properties dictate the performance of the network. Even for such a basic setup, we found that accurate analysis is quite challenging. Moreover, the introduction of additional features, such as allowing the source-destination pair to exhibit mobility seems to make the analysis significantly more challenging. Proceeding from this preliminary model, this chapter introduces a more powerful approach for the steady-state performance analysis of SPMANETs, presenting a generalized framework.

In this chapter, we restrict ourselves again to the single-unicast case, where a single source node transmits packets to a destination using mobile relays. However, we relax the constraint that the source/destination nodes must be fixed. Moreover, we now present a framework that is applicable to all classes of mobility models which exhibit statistical time stationarity. It is noteworthy that this general class of mobility models encompasses the commonly used ones such as random waypoint, random walk, and Gauss-Markov mobility models. Also, the communication range of the node can be varied – doing so in the methodology presented in the previous chapter would again necessitate a large increase in the Markov-chain complexity. In itself, this setup presents a complex scenario when one takes into consideration practical constraints such as limited node storage, random contact times, and contention between nodes. We proceed to show not only that such analysis is feasible, but also that

it can be achieved with accuracy. We achieve this by means of a two-fold approach which combines 1) Embedded Markov-Chain identification, and 2) Chain reduction by state collapsing. We then illustrate the framework in detail for the random-walk mobility model. Finally, we validate the analysis using simulations for three different mobility models: the random waypoint, and the restricted random waypoint models, in addition to the random-walk model. The results pertaining to this chapter are available in our papers [71–73].

The rest of this chapter is organized as follows. In Section 5.1, we detail the network model used in developing the generalized analysis framework. In Section 5.2, we discuss and detail the evolution of the analytical framework, employing queuing-theoretic tools for Markov-chain complexity reduction. We then discuss the methodology in detail for the single-unicast case employing two-hop routing. In Section 5.4, we discuss the analysis and simulation results for the above scenario. We then summarize our findings in Section 5.5.

5.1 Network Model

5.1.1 Definitions

Whenever two nodes are within communication range of each other, we say that a “contact” has occurred between them. However, they may only communicate when a “link” exists between them. The rules for establishing a link such that contention is avoided is described later in this section. A “block” of packets, say q in number, can be transmitted through any link in one epoch. Hence q is related to the transmission bandwidth in the network.

Throughout the chapter, we assume the following communication scenario: Several nodes are deployed in a confined region, which then move independently according to a certain mobility model. We assume that the nodes are identical i.e., they have the same storage buffer size, communication range, and mobility parameters. These

nodes are designated as “relay” nodes and are n in number with a buffer space of B blocks each. The network also includes two nodes identified as the “source” and “destination” respectively, which are also mobile, and have unlimited storage capacity. Further, nodes may have a certain finite communication range within which it is able to send or receive packets from another node. We consider our network to be in discrete time, where time is sliced up into several epochs. At every epoch, a single block may be transmitted/received by any node. The last two assumptions are made for clarity and ease of notations, and does not cause any loss of generality. Further, a small buffer size is justified since devices use less memory space for communication, though they might have larger storage capacity. We initially assume loss-free channel and later show how we can incorporate channel losses into consideration.

5.1.2 Interference Model and Contention Resolution

For the duration of communication between two nodes, others within communication range of either node must remain silent i.e., neither send nor receive packets. This is ensured in order to avoid hidden-terminal and exposed-terminal situations. Moreover, the source/destination node tries to establish a fresh link at each epoch, for which several relay nodes may contend. The contention scenario is resolved as follows: At the beginning of each epoch, each relay node advertises its own ID, advertising that a link be established between itself and any source/destination node within communication range. A source(or destination) node, if free, will collect all such requests (within a window of time that is negligible compared to the total duration of the epoch), and select a candidate at random with equal probability, and establishes a link with the same. Further, it is assumed that the source node always prioritizes establishing a link with the destination, if possible, over linking with any relay node.

5.1.3 Mobility Model

Throughout this work, the underlying mobility model is said to exhibit “stationarity”. This means that the probability distribution of the “state” of a node’s motion converges over time to a fixed distribution. Though this is critical to our analysis, it is not an unreasonable assumption. It is well-known that a mobility model is meaningful for network evaluation only if it exhibits stationarity [74]. In particular, it is obvious that the random-walk, the random-waypoint model, and their variants are known to be stationary under proper choice of parameters. The mobility of any node v is denoted by a random process $\chi_v(t)$, which at each instant t is a probability distribution on the state-space \mathcal{S}_{mob} of the given mobility model. For clarity, it is assumed that the state-space \mathcal{S}_{mob} is discrete. Each state may include information such as position, velocity, and waypoint location, etc. Typically, one can describe the state transitions as a linear relationship by means of a transition function $\Psi_{mob}(\cdot)$ (given by the model) as follows. Let $\mathbf{p}(t)$ be the probability distribution of the node’s mobility state at time t . Also, let the mobility model have a memory of m' time-steps, where m' is a positive integer. Then, $\mathbf{p}(t+1) = \Psi_{mob}[\mathbf{p}(t), \mathbf{p}(t-1), \dots, \mathbf{p}(t-m')]$ for $t > m'$. For example, in case of the random-walk mobility model, $\Psi_{mob}(\cdot)$ simply involves the multiplication of $\mathbf{p}(t)$ by a constant state-transition matrix to obtain $\mathbf{p}(t+1)$. Since the mobility model is assumed to be stationary, it has a unique steady-state probability distribution, $\boldsymbol{\pi}_{mob}$ which satisfies $\boldsymbol{\pi}_{mob} = \Psi_{mob}[\boldsymbol{\pi}_{mob}, \dots, \boldsymbol{\pi}_{mob}]$.

5.1.4 Two-Hop Single-Copy Routing

Messages at the source node are packetized, and the source tries to transmit them to one particular destination. Throughout this work, we assume that the source node never transmits multiple copies of the same packet. Hence, at any given time, there is at most one copy of the packet in the network. In our recent work [75] in the context of Delay Tolerant Networks, we have shown that single-copy routing can indeed be

preferable to multi-copy routing especially in scenarios where messages can be sent by using erasure codes. A mobile (relay) node, accepting packets from the source node whenever a link is established with them, retains the each packet until a link with the particular destination node occurs. No relay-to-relay communication occurs. In addition to the above, direct source-destination links, whenever they occur, are also exploited.

5.2 *Details of the Analysis*

5.2.1 Notations

Relay nodes in the network is identified by a unique integer, $v \in \{1, 2, \dots, n\}$. For a Markov Chain $X(t)$ with state-space Ω , the steady-state probability of any state $x \in \Omega$ is denoted by $\pi(x)$. The vector $\boldsymbol{\pi}$ denotes the steady-state distribution for the entire state-space \mathcal{S} .

5.2.2 Overview of the Concept

Our methodology for throughput computation is as follows: We first identify an embedded Markov-Chain which describes the exact dynamics of communication, and show that by identifying certain “desirable” states of the network and analyzing their steady-state probability distributions, one can compute the throughput. The analysis is further facilitated by a novel technique called “chain collapsing” which drastically reduces the complexity of the underlying chain. We devote the rest of the discussion to developing the above steps into a generalized framework for performance analysis in SPMANETs.

The state of any node k at time t in the network is designated by an ordered pair consisting of mobility-state and its buffer-occupancy, as $(\chi_k(t), b_k(t))$. In other words, at any epoch t , $\chi_k(t) \in \mathcal{S}_{mob}$, $0 \leq b_k(t) \leq B$ at any time t for any node k . Next, we define the state of the entire network as the $2(n+1)$ -tuple $\mathcal{Y}(t) \triangleq (\chi_1(t), \dots, \chi_n(t), b_1(t), \dots, b_n(t), \chi_s, \chi_d)$. Here, χ_s and χ_d are the mobility states of

the source and the destination. The transitions of the process $\mathcal{Y}(t)$ within its state-space can be determined from the mobility model and the underlying communication protocols. With the above state-space description of the network, we define the achievable throughput as the *expected rate at which packets are transferred s to d when the network is in steady-state*. In other words if $\mathcal{N}_{s,d}(\tau)$ packets are transmitted from s to d in time τ , the throughput capacity is given by the relation $\mathcal{C}_{s,d} = \frac{\mathcal{N}_{s,d}(\tau)}{\tau}$.

In essence, we have to analyze the complex multi-dimensional random process $\mathcal{Y}(t)$ in order to derive the throughput for the SPMANET network model. Clearly, the full state-space description of the network, consisting of $\Theta(B^n |\mathcal{S}_{mob}|^n)$ states, is prohibitively large to work with. However, we develop a methodology to drastically reduce the state-space of this process.

However, we develop a methodology to drastically reduce the state-space of this process.

5.2.3 Embedded Markov Chain and Chain-collapsing

In order to simplify the analysis, the first step is to identify certain symmetries in the network that simplifies the state-space. For example, in a scenario where relay nodes are identical, the throughput contribution per relay node will be the same for all relays. As a result, we can view the state of the network from a single relay-node's perspective, compute the throughput due to that node, and scale it up by n . However, this would still leave us with $\Theta(B |\mathcal{S}_{mob}|^{n+2})$ states, since the contention effects due to the other nodes' mobility have to be fully accounted for. However, one can perform the analysis using a less cumbersome alternative means, without additional assumptions and approximations, by drastically reducing this state-space. We do so as follows: (1) Derive a Markov Chain from $\mathcal{Y}(t)$ such that the steady-state probability distribution and the transition probabilities of $\mathcal{Y}(t)$ at steady-state are preserved. (2) For a particular relay node v , identify all the “desirable” states in

which packets are sent to the destination by v , and certain additional “auxiliary”, such that one can construct an “embedded” Markov chain with the combined set. As a result, we will have the following:

Proposition 1 *Let σ_i be the desirable states in the embedded Markov Chain for a particular relay node v . Also, let η be the average time that the process $\mathcal{Y}(t)$ spends in either the desirable or the auxiliary states of v . Then, the throughput contribution due to node v is given by $\mathcal{C}_{v,s,d} = \eta \sum \pi(\sigma_i)$.*

Next, we note that the embedded Markov chain described thus is still has too many states, as one needs to track the movement of all the nodes in order to fully account for contention effects. We will now summarize the novel “Chain Collapsing” approach that we have developed, simplifying the chain further. The idea is to construct a much smaller chain defined with subsets of states in the primal chain as states in the new chain, as shown in Fig. 34, such that the total steady-state probabilities of individual subsets in the former are same as the corresponding steady-state values in the latter. We then partition the states of the embedded chain such that no subset has both kinds of states (desirable and auxiliary). Then, from Proposition 1, it is clear that the knowledge of steady-state distribution for the collapsed chain constructed with such a partition is enough to compute the throughput. We summarize the process in the following lemma:

Lemma 3 (*Chain Collapsing*) *Let \mathcal{M} be an irreducible Markov chain with a set of states denoted by Ω , with a steady-state distribution $\boldsymbol{\pi}(\Omega)$ for its states. Let $\{\Omega_i\}_{i=1}^s$ be disjoint subsets of Ω such that $\bigcup_{i=1}^s \Omega_i = \Omega$. Then, a new Markov chain defined with $i = 1 \cdots s$ corresponding to each of the above subsets, with transition probabilities corresponding to the “subset-averaged” values of those from the original Markov chain \mathcal{M} , has a steady-state distribution $\boldsymbol{\pi}'$ such that $\pi'(\Omega_i) = \sum_{j \in \Omega_i} \pi(j)$. Moreover, the*

transition probabilities for the new chain are given by the following relationship:

$$\theta'_{\Omega_r, \Omega_l} = \sum_{j \in \Omega_r} \sum_{k \in \Omega_l} \theta_{jk} \pi(j | \Omega_r) = \frac{1}{\pi'(\Omega_r)} \sum_{j \in \Omega_r} \sum_{k \in \Omega_l} \theta_{jk} \pi(j),$$

where $\theta_{j,k}$ for all $j, k \in \Omega$ are the transition probabilities of the old chain \mathcal{M} .

Proof: Let $\{\theta_{ij}\}$ be the probability transition matrix of the original Markov chain \mathcal{M} . Let E' be the edge-set of the modified Markov chain constructed from the subsets. For two given subsets of states Ω_l and Ω_m we say $(\Omega_l, \Omega_m) \in E'$ iff $\exists a \in \Omega_l, a' \in \Omega_m$ such that $\theta_{aa'} \neq 0$. A sample state diagram is shown in Fig. 34. Then, we have the following:

$$\pi'_{\Omega_l} = \sum_{k \in \Omega_l} \pi(k) = \sum_{k \in \Omega_l} \sum_{j \in \Omega} \theta_{jk} \pi(j) \quad (56)$$

$$= \sum_{r=1}^s \sum_{j \in \Omega_r} \sum_{k \in \Omega_l} \theta_{jk} \pi(j) = \sum_{r=1}^s \pi'(\Omega_r) \frac{1}{\pi'(\Omega_r)} \sum_{j \in \Omega_r} \sum_{k \in \Omega_l} \theta_{jk} \pi(j) \quad (57)$$

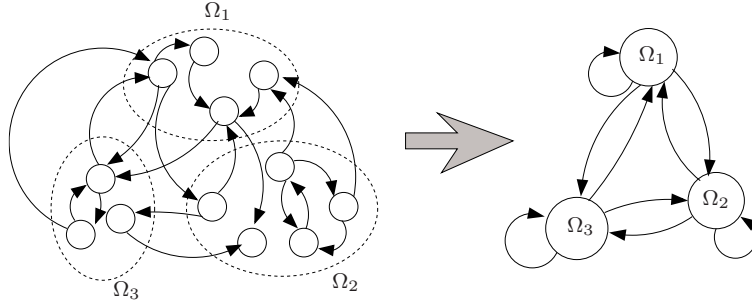


Figure 34: Collapsing a Markov Chain

Now, let us define $\theta'_{\Omega_r, \Omega_l} \triangleq \sum_{j \in \Omega_r} \sum_{k \in \Omega_l} \theta_{jk} \pi(j | \Omega_r)$. We call these the “subset-averaged transition probabilities”. Then, clearly, the final expression in (57) becomes $\sum_{r=1}^s \pi'(\Omega_r) \theta'_{\Omega_r, \Omega_l}$, since $\pi(j | \Omega_r) = \frac{\pi(j)}{\pi'(\Omega_r)}$. Therefore, we have an “equivalent” Markov chain with fewer states, and we are interested in only the sum of steady-state probabilities of a certain subset of states. Further, the transition probabilities for the new chain can be constructed by taking the subset-averaged values of the corresponding transition probabilities of the original chain. To summarize the above lemma, we note

that chain collapsing enables us to extract only the necessary information from the original Markov chain. In particular, we first need to reduce the throughput computation problem to computing the total steady-state probabilities of certain subsets of a well-defined embedded Markov chain. Note that individual steady-state probabilities of states within one particular subset are of no interest to us. In addition, we only require the “steady-state” transition probabilities between the subsets (notated as $\theta'_{\Omega_r\Omega_l}$ in the lemma) to compute the steady-state probabilities of Ω_i . The key is to derive $\theta'_{\Omega_r\Omega_l}$ in an alternative manner rather than from the original chain. We show that this is possible in our network model. Hence, even though we lose some information regarding the dynamics of the original chain, the new chain has enough information to compute the exact throughput.

The rest of the analysis involves the computation of the transition probabilities $\theta'_{\Omega_r\Omega_l}$, followed by computation of the actual steady-state probabilities of Ω_i using the collapsed chain. The details vary from one communication scenario to the other.

5.3 *Analysis of Single Source-Destination Unicast*

We start by defining the following embedded Markov chain:

- S_l , with $1 \leq l \leq B$ is the set of possible network states wherein the *most recent* link that node v was involved in was with the source, resulting in l blocks of q packets in the buffer after communicating with the latter. Similarly, define $D_{l'}$, for all $B - 1 \geq l' \geq 0$ to account for communication with the destination.
- F is the set of possible network states wherein the most recent link that node v was involved in was with the source, but v was unable to communicate with the latter due to lack of enough buffer space (i.e., Full/saturated buffer condition). Similarly, define E to account for the case when a link is established with the destination when the buffer-occupancy of v is zero.

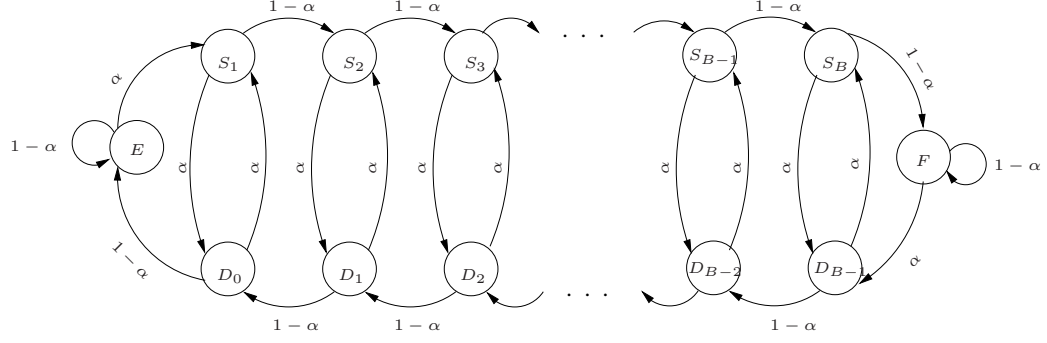


Figure 35: Collapsed Markov chain for two-hop single unicast

The set D_l contain the desirable states, whereas S_l , E , and F consist of auxiliary states. Applying Lemma 3 on the above Markov chain, we arrive at the collapsed chain for our SPMANET network-model. We still need to compute the transition probabilities by taking node mobility and contention-related issues into account. To do so, we first observe that the state transitions from any state S_l , D_l , F or E depends only on the node mobility, and not on the buffer occupancy of any node, since the contention model that we assumed is independent of the relay nodes' buffer occupancies. The resulting Markov chain with appropriately designated transitions resembles a Quasi-Birth-Death (QBD) queue-process, as depicted in Fig. 35. One can repeat this process with minor modifications for contention models dependent on buffer-occupancy (which we avoided for clarity).

5.3.1 Contention Analysis and Derivation of Mobility Parameters

In order to complete the analysis of throughput, we need to determine the parameter α in the collapsed chain shown in Fig. 35. First, we note that α depends on the mobility model, number of nodes, and on the contention model. We derive the same in the following way: establish the dependence on the contention protocol first, and finally establish the dependence on the mobility characteristics. The first step is accomplished with the aid of yet another embedded chain, defined on the following sub-sets of states, defined with respect to the particular relay-node v :

- S_F, D_F : Most recent contact occurred with the source (destination), and v won contention
- S_E, D_E : Most recent contact occurred with the source (destination), and v lost contention

Once again, one may collapse these subsets into just four states, resulting in the new Markov chain. Let us call this the *four-state contention chain*. We note that the quantity α in the original chain in Fig. 35 can be derived from the four-state contention chain as the probability that the latter, starting at state S_F , goes through the state D_F at least once before reaching S_F again. This probability is in-turn computed using the *ergodic fundamental matrix* of the four-state contention chain. The details are discussed further when we discuss the corresponding issues under multiple unicast. Proceeding thus, we can compute α as follows:

Lemma 4 *Let α_0 be the probability that a node currently in contact with the source (or destination) will have a contact with the destination (or source) before coming in contact with the former again. Let β_c be the average probability that a node loses contention on meeting the source/destination node. Then, we have $\alpha = \frac{\alpha_0}{2\alpha_0\beta_c + (1-\beta_c)}$.*

A useful result that enables the computations of α_0 and β_c is as follows. This is a particular case of the Arrivals that See Time Averages (ASTA) theorem [70] in general queuing theory.

Lemma 5 *Let π_{mob} be the mobility-state probability distribution at steady state for a node a_3 , and let $\pi_{mob}^{(a_1, a_2)}$ be the distribution of the mobility states of a_3 whenever a contact between a certain node pair (a_1, a_2) occurs. Then the two distributions will be identical, i.e., $\pi_{mob} = \pi_{mob}^{(a_1, a_2)}$.*

Hence, the location of the “other” relay nodes and the destination (source) node, whenever the relay is in contact with the source (destination), is independent of all

network dynamics. Hence, the same quantities α and $1 - \alpha$ characterize all the state transitions in the chain shown in Fig. 35.

Our derivation of the parameters α_0 and β_c in terms of known mobility parameters is summarized by the following lemma:

Lemma 6 *Let T_0 be a random variable representing the inter-contact duration (i.e., given that a pair of nodes are in contact in the current epoch, the time until the next contact occurs between the same pair), and let T_∞ be the random variable representing the waiting time until two nodes meet, given that they are distributed according to the steady-state spatial location distribution. Let \mathbf{x}' set of mobility states (i.e., $\mathbf{x}' \subset \mathcal{S}_{mob}$) such that a node currently in any state of \mathbf{x}' is within communication range from a node in state $\mathbf{x} \in \mathcal{S}_{mob}$. Let π_{spt} be the steady-state spatial-location distribution due to the underlying mobility model. Then we have*

$$\begin{aligned}\alpha_0 &= \sum_{\tau=1}^{\infty} F_{T_\infty}(\tau) p_{T_0}(\tau) \\ 1 - \beta_c &= \left(1 - \frac{1}{\mu_0}\right) \sum_{k=0}^{n-1} \sum_{\mathbf{x}} \frac{\pi_{spt}(\mathbf{x})}{k+1} \binom{n-1}{k} \pi_{spt}^k(\mathbf{x}') \{1 - \pi_{spt}(\mathbf{x}')\}^{n-1-k}\end{aligned}$$

where μ_0 is the mean inter-contact time.

Proof: First, we derive α_0 . Consider the network at steady-state dynamics. Let us assume a particular relay and the source are in contact at time $\tau = 0$. Clearly, T_0 is the random variable representing the time until the next contact between the particular relay node and the source occurs. From Lemma 5, one observes that the random location of the destination node, sampled at each instance where the source and a particular relay node are in contact, follows the steady-state spatial distribution of the mobility model. Hence, T_∞ is the random variable representing the expected waiting time until the particular relay and destination come in contact. Further, it may be deduced that T_0 and T_∞ are independent since the probability distribution of either one does not change with any information regarding the other. Hence, the

required probability α_0 may be expressed as:

$$\begin{aligned}\alpha_0 &= P(T_\infty < T_0) = \sum_{\tau=1}^{\infty} P(T_0 = \tau) P(T_\infty < \tau | T_0 = \tau) \\ &= \sum_{\tau=1}^{\infty} F_{T_\infty}(\tau) p_{T_0}(\tau)\end{aligned}$$

Next, we derive β_c . The quantity $1 - \beta_c$, which denotes the probability that a relay node wins contention on contact with a source/destination, is the probability that the source and destination nodes are not in communication range and the particular relay node wins contention from competing relay nodes. One can see that by ergodicity, the probability that the source and destination are in contact is given by $\frac{1}{\mu_0}$ where μ_0 is the mean inter-contact time for the given mobility model. The probability that a relay node wins contention in the presence of competing relays and in the absence of destination can be analyzed as follows: Let \mathbf{x} be the location of the particular relay node. Let \mathbf{x}' be the set of neighboring locations. By Lemma 5, ergodicity, and by the independence of node mobilities, one can see that the number of competing relay nodes follows the Binomial distribution with parameters $n - 1$ and $\pi_{spt}(\mathbf{x}')$. Further, given that there are k competing relay nodes, the probability of winning contention is $\frac{1}{k+1}$. Then, the probability p_{cr} that the particular relay node wins contention with competing relays is given by:

$$p_{cr} = \sum_{k=0}^{n-1} \binom{n-1}{k} \pi_{spt}^k(\mathbf{x}') \{1 - \pi_{spt}(\mathbf{x}')\}^{n-1-k} \frac{1}{k+1}$$

Averaging the above over the steady-state spatial distribution of the relay node's location \mathbf{x} , we have:

$$p_{cr} = \sum_{k=0}^{n-1} \sum_{\mathbf{x}} \binom{n-1}{k} \pi_{spt}^k(\mathbf{x}') \{1 - \pi_{spt}(\mathbf{x}')\}^{n-1-k} \frac{\pi_{spt}(\mathbf{x})}{k+1}$$

Proceeding further, one can derive the distribution of T_0 in terms of the distribution of T_∞ . Hence, we have complete information that determines α_0 uniquely. All we need is to just compute the moments $E[T_0^k]$ in terms of known parameters. The

former can be accomplished by first looking at the moments of the distribution of T_∞ , $E[T_\infty^r]$:

$$\begin{aligned} E[T_\infty^r] &= \sum_{t=1}^{\infty} (t-1)^r p_{T_0}(t-1) \\ &= E[T_0] \sum_{t=0}^{\infty} (t-1)^r P[T_0 \geq t] \end{aligned}$$

For the last step, see Section 5.1 of Chapter 2, [76]. Reordering the terms,

$$\begin{aligned} E[T_0]^{-1} E[T_\infty^r] &= \sum_{m=1}^{\infty} \left(\sum_{l=0}^{m-1} l^r \right) P[T_0 = m] \\ &= \sum_{m=1}^{\infty} \frac{1}{r+1} \sum_{k=0}^r \binom{r+1}{k} B_k m^{r+1-k} P[T_0 = m] \end{aligned}$$

where B_k are the Bernoulli numbers. Hence,

$$E[T_0]^{-1} E[T_\infty^r] = \frac{1}{r+1} \sum_{k=0}^r \binom{r+1}{k} B_k E[T_0^{r+1-k}]$$

Proceeding further, one may obtain the *exponential generating function* of the moments of T_0 in terms of that for T_∞ as:

$$\sum_{l=1}^{\infty} \frac{x^l}{l!} E[T_0^l] = E[T_0] (e^x - 1) \sum_{r=0}^{\infty} E[T_\infty^r] \frac{x^r}{r!} \quad (58)$$

The distribution of T_∞ is well-approximated by a geometric distribution (the discrete-time equivalent of exponential distribution), when the communication range is small compared to the deployment area. This is a standard approximation in Markov-chain theory and one is referred to Section 5.4 in Chapter 3 of [76] for a proof of the same. Then, the only quantity to be determined would be the mean of the distribution of T_∞ . We provide an approximate expression for the same in the random-walk-on-grid case. For other mobility models, deriving this parameter is a separate exercise, and is beyond the scope of our discussion.

To finish up the analysis, one can easily verify that the steady-state probabilities of each of the “desirable” states in Fig. 35 is $\frac{\alpha}{2}$ each, and that of the states E and

F are $\frac{1-\alpha}{2}$ each. One can then verify that the fraction of successful links with the destination that results in the successful transmission of a block to the destination, is given by $\frac{B}{B+2\beta_c-1+(1-\beta_c)\alpha_0^{-1}}$. In addition, successful links occur once in $\mu_0(1-\beta_c)^{-1}$ epochs on the average. Hence, the throughput capacity of the network is given as shown in the theorem, completing our analysis for the single-unicast case.

Theorem 5 *For an SPMANET with n relay nodes and a buffer size of B with an arbitrary stationary mobility model, let*

$$1 - \beta_c = \frac{1}{\mu_0} + \left(1 - \frac{1}{\mu_0}\right) \sum_{k=0}^{n-1} \sum_{\mathbf{x} \in \mathcal{S}_{mob}} \frac{\pi_{spt}(\mathbf{x})}{k+1} \binom{n-1}{k} \pi_{spt}^k(\mathbf{x}') \{1 - \pi_{spt}(\mathbf{x}')\}^{n-1-k}$$

and $\alpha_0 = \sum_{\tau=1}^{\infty} F_{T_{\infty}}(\tau) p_{T_0}(\tau)$, where π_{spt} is the steady-state spatial node-distribution, \mathbf{x}' be the set of mobility states that are within communication range of states \mathbf{x} , $F_{T_{\infty}}$ be the c.d.f. of the waiting time for any contact at steady-state, and p_{T_0} is the p.d.f. of the inter-contact time and μ_0 be its mean. For such a network, the two-hop throughput capacity for a single mobile source-destination pair is given by:

$$\mathcal{C}_{s,d} = \frac{n(1-\beta_c)}{\mu_0} \frac{B}{B+2\beta_c-1+(1-\beta_c)\alpha_0^{-1}} + \frac{1}{\mu_0}$$

The second part of the expression ($\frac{1}{\mu_0}$) is the throughput contribution due to direct source-destination contacts. Concluding this section, we note that the above analysis shows the following.

- A. The throughput capacity of the network can be computed from the network purely from the knowledge of three of its parameters, μ_0 , β_c , and α_0 which have been characterized above. In other words, complete knowledge of mobility statistics may not be necessary.
- B. For certain mobility models, it might only be feasible to approximate the above three parameters, or in certain cases may only be obtained from simulations. However, obtaining these parameters by simulation is far more efficient than

simulating the entire network to study its throughput behavior. Nevertheless, a full network-level simulation is far less insightful. Further, the feasibility of obtaining these parameters analytically is dependent on the mobility model, and does not affect the validity of our throughput model.

5.4 Analysis and Simulation Results

Here, we show that our framework to compute the throughput capacity can be applied to any mobility model that exhibits time-stationarity. As examples, we consider three mobility models often used in literature.

5.4.1 Case 1: Random Walk Model

In this mobility model, the deployment area consists of a $M \times M$ square grid. Nodes move independently according to a random walk process on the grid. This mobility model is followed by each of the relay nodes, and also by the source and destination nodes. The communication range of each node is such that it can establish a link with any node which is R grid points away. The state transition matrix of the random walk model can be chosen such that the steady-state spatial distribution of node location will be uniform, which in this case would be $\frac{1}{M^2}$ for each grid point. We will derive the expression for throughput for the case when the communication range of the node is small compared to the deployment area. The grid-mobility model is illustrated in Fig. 36.

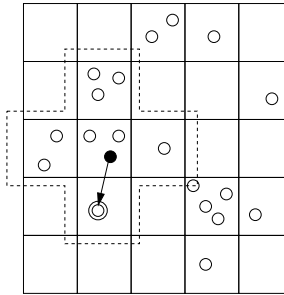


Figure 36: Random-Walk mobility model for $R = 1$. Note: solid bubble = sending; concentric bubble = receiving; hollow bubble = idle

Lemma 7 *For $R \ll M$, the expected inter-contact time for the random-walk model is given by $\mu_0 \approx \frac{M^2}{(R+1)^2}$.*

Proof: To derive the expected inter-contact time μ_0 , one considers the Markov chain constructed by the pair $(\chi_1(t), \chi_2(t))$, a random process corresponding to the positions of two independent nodes. Defining A to be the subset of states wherein $\chi_1(t)$ and $\chi_2(t)$ are within communication range, one can see from Kac's formula (Chapter 2, page 21 in [76]) that $E[T_0] = \frac{1}{\pi_{spt}(A)}$ where $\pi_{spt}(A)$ is the sum of steady-state probabilities of the states in A . One then observes that $\pi_{spt}(A)$ is $\frac{1}{M^2}$ times the average number of grid points in communication range, and the latter is approximated well by $(R+1)^2$ for the case when $R \ll \sqrt{M^2}$.

The contention failure probability can be directly derived from Lemma 6) as

$$1 - \beta_c = \frac{(R+1)^2 M^2}{n} \left[1 - \left\{ 1 - \frac{(R+1)^2}{M^2} \right\}^n \right] \quad (59)$$

The above is derived as a direct trivial result of Lemma 6, and from the fact that the steady-state spatial distribution for this grid-based mobility model is uniform. Further, from Kac's formula (See Section 5.1, Chapter 2 of [76]), $\pi(\mathbf{x}') = \frac{(R+1)^2}{M^2}$ independent of \mathbf{x}' .

Finally, one can derive the parameter α_0 from the known distribution functions of T_0 and T_∞ for the random walk model as:

$$\alpha_0^{-1} \approx 1.23(R+1)^2 \log M \quad (60)$$

To derive the above formula, one first notes that T_∞ is well-approximated by a geometric distribution (the discrete-time equivalent of exponential distribution) that takes positive integer values. To see this is true, one is referred to Section 5.4 in Chapter 3 of [76]. The derivation of the mean for this distribution is beyond the scope of this chapter, but it can be shown to be approximately $\gamma M^2 \log M$ where γ is close to 0.615. Let the probability-of-success for the geometric distribution followed

by T_∞ be p' . We know that the mean of T_∞ , given by $\frac{1}{p'}$ has value $\gamma M^2 \log M$ in terms of the network parameters. Hence, we know p' to be $(\gamma M^2 \log M)^{-1}$. From Lemma 6,

$$\begin{aligned}\alpha_0 &= \sum_{\tau=1}^{\infty} F_{T_\infty}(\tau) p_{T_0}(\tau) = \sum_{\tau=1}^{\infty} (1 - (1 - p')^\tau) p_{T_0}(\tau) \\ &= 1 - \sum_{\tau=1}^{\infty} (1 - p')^\tau p_{T_0}(\tau) = 1 - EGF_{T_0} \{\log(1 - p')\}\end{aligned}\tag{61}$$

where EGF_{T_0} denotes the exponential moment-generating function of the random variable T_0 . Hence, it is sufficient to compute the moments of the random variable T_0 to complete the derivation. However, it is also necessary to show that $\log(1 - p')$ is well within the region of convergence of EGF_{T_0} . Both tasks can be done as follows, using the result of Section 5.1, Chapter 2, of [76] and using the fact $E(T_0) = \frac{M^2}{(R+1)^2}$:

$$\begin{aligned}E(T_\infty^r) &= \sum_{t=1}^{\infty} (t-1)^r p_{T_\infty}(t-1) \\ &= (R+1)^{-2} M^2 \sum_{t=1}^{\infty} (t-1)^r P(T_0 \geq t) \\ \Rightarrow \frac{(R+1)^2}{M^2} E(T_\infty^r) &= \sum_{m=1}^{\infty} \left(\sum_{l=0}^{m-1} l^r \right) P(T_0 = m) \\ &= \sum_{m=1}^{\infty} \frac{1}{r+1} \sum_{k=0}^r \binom{r+1}{k} B_k m^{r+1-k} \\ &\quad \times P(T_0 = m) \\ &= \sum_{m=1}^{\infty} \sum_{k=0}^r \binom{r+1}{k} B_k E(T_0^{r+1-k})\end{aligned}\tag{62}$$

In the above, B_k are the Bernoulli numbers, and $B_0 = 1$. Its exponential generating function is known to be $\frac{x}{e^x - 1}$, with ROC $\|x\| < 2\pi$. Proceeding further, one may obtain the *exponential generating function* of the moments of T_0 in terms of that for T_∞ as:

$$\sum_{l=0}^{\infty} \frac{x^l}{l!} E[T_0^l] = 1 + \frac{M^2 (e^x - 1)}{(R+1)^2} \sum_{r=0}^{\infty} E[T_\infty^r] \frac{x^r}{r!}\tag{63}$$

Using the exponential generating function for the geometric-distributed random variable T_∞ (which has ROC $x < -\log(1 - p')$), we will have:

$$EGF_{T_0}(x) = 1 + \frac{M^2(e^x - 1)}{(R + 1)^2} \frac{p'}{1 - (1 - p')e^x} \quad (64)$$

which has ROC $x < \min(2\pi, -\log(1 - p'))$. Clearly, $\log(1 - p')$ is within the ROC since it is negative, and p' is quite small for the former to exceed 2π in absolute value. Hence, we can use (61) to compute α_0 . Using (64) in (61) and simplifying, we will have the required expression for α_0 as in (60):

$$\begin{aligned} \alpha_0 &= \frac{1}{2} p' M^2 (R + 1)^{-2} \\ &= \frac{1}{2(R + 1)^2 \gamma \log M} \end{aligned}$$

Using (59), (60) and the result of Lemma 7 in Theorem 5, one will obtain the throughput capacity of a network under the random-walk model in closed form.

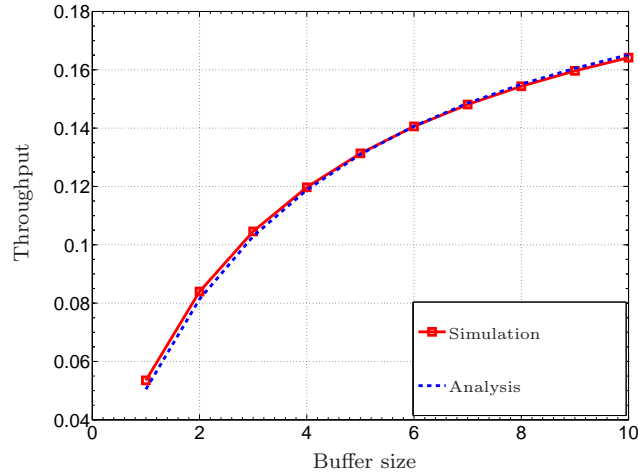


Figure 37: Throughput vs buffer size (Random Walk)

Mobile networks exhibiting random-walk mobility with real-time packet transmissions were simulated in MATLAB, and the results have been plotted in Figs. 37, 38, and 39. The variation of throughput with the node buffer size is shown in Fig. 37.

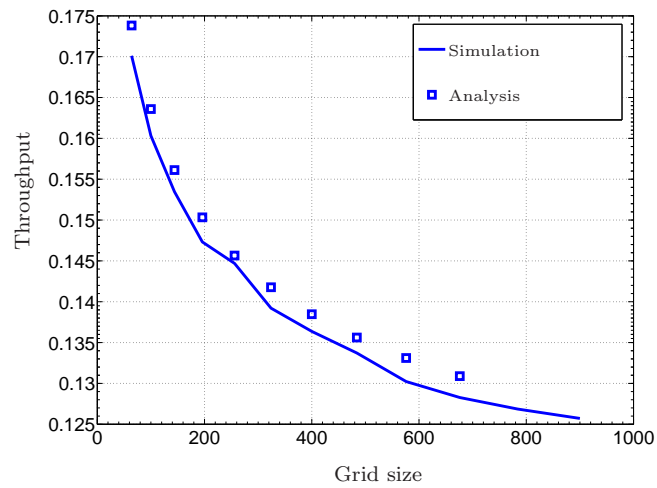


Figure 38: Throughput vs grid size (Random Walk)

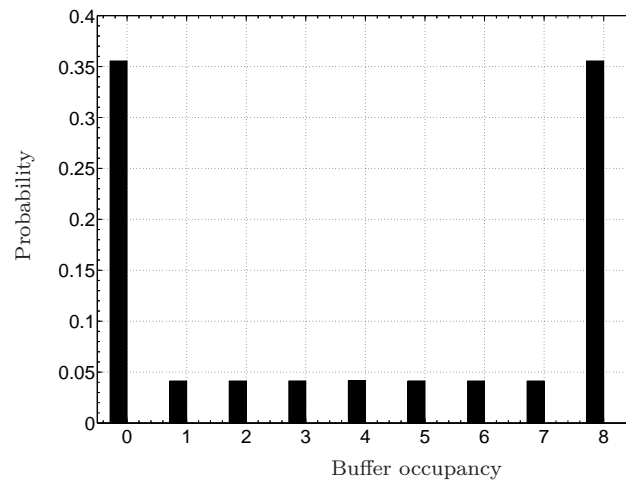


Figure 39: Steady-state buffer distribution (Random Walk)

Here, the node buffer size was made to vary from 1 to 10, while the number of nodes was kept at 100 and the grid size was maintained as 20×20 . In the second case (Fig. 38), the grid-size was varied from 8×8 to 30×30 , while keeping the density of nodes and buffer size to be constant at 0.5 per grid point and 8 packets, respectively. In both cases, the communication range of the nodes R was chosen to be three grid points wide. The packet-loss probability ϵ was chosen to be 10%. The accuracy of our analysis is evident from the fact that the plots show negligible error.

An interesting consequence of two-hop communication is illustrated in Fig. 39. Here, it is seen that relay nodes spend considerably higher times in the empty buffer state, which is detrimental to the throughput capacity. It is then evident that a considerable improvement in the throughput is to be gained if we allow relay-relay communication using a back-pressure policy, where a node with higher buffer occupancy transfers packets to a node with a lower occupancy. However, the improvement in the throughput comes at the cost of spending more node energy. The extension of analysis for multi-hop is more complicated and is under investigation.

5.4.2 Case 2: Random Waypoint Model

The random waypoint (RWP) mobility model is often used in simulation studies of real-world networking protocols [77]. Under this model, each node choose a random location (called “waypoint”) in the deployment area, and move towards the same with a randomly chosen speed. Upon reaching the waypoint, they wait for a random interval of time, after which the next waypoint and speed are chosen and the process is repeated.

The first step to compute the throughput is to obtain the three parameters mentioned in Theorem 5. Though it has been shown that approximate analytical expressions can be obtained for some basic characteristics of RWP [78, 79], the same for the parameters that we require in our analysis have not been obtained in closed form,

to the best of our knowledge. As an example, good approximations are available for the steady-state spatial node distribution, and can be used to compute β_c using (6) efficiently. However, it is hard to obtain an analytical expression for the same due to the highly non-uniform nature of the steady-state spatial node density. Nevertheless, the other parameters can be obtained numerically by a quick simulation involving just three nodes.

A plot of the comparison between the throughput obtained using a full network-level simulation, with the one obtained using our framework is shown in Fig. 40, for varying buffer size. For our simulation, the deployment area was a $5\text{ km} \times 5\text{ km}$ square region wherein 100 nodes were deployed. The node velocity was chosen from a uniform random distribution, with $v_{min} = 3\text{ m/s}$ and $v_{max} = 10\text{ m/s}$. The communication range was chosen to be 250 m . The pause-time was modeled as an exponential distribution with a mean of 60 s . The node buffer size at each was chosen to be 8.

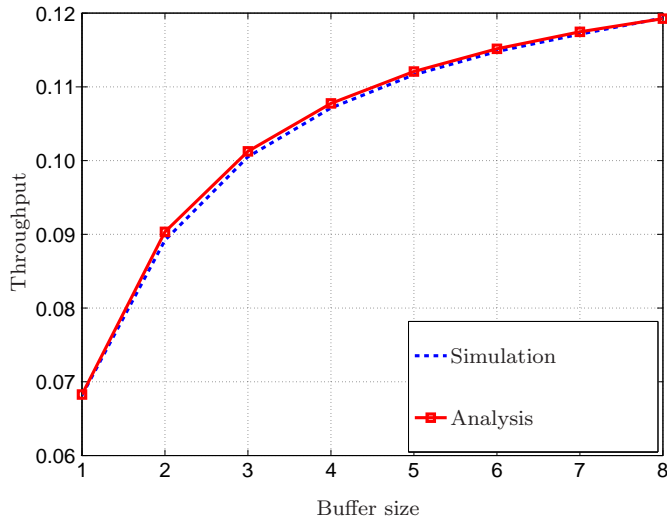


Figure 40: Throughput vs buffer size for Random Waypoint Mobility

5.4.3 Case 3: Restricted Random Waypoint Model

A restricted version of the random waypoint model is sometimes used [80]. We use one version of this model here, wherein ten sub-areas are designated in the overall

deployment region. Nodes choose their waypoints within the same sub-area they are currently in, with a high probability. Occasionally, it may choose a waypoint in some other area and move to the same freely. For our study, the same parameters as in the unrestricted Random Waypoint model were chosen, in addition to fixing the probability of transition to a different area to be 0.05. Again, the three parameters required for throughput computation were obtained numerically. From the results of the simulation, plotted in Fig. 41 it is once again evident that our framework is quite versatile for various mobility models.

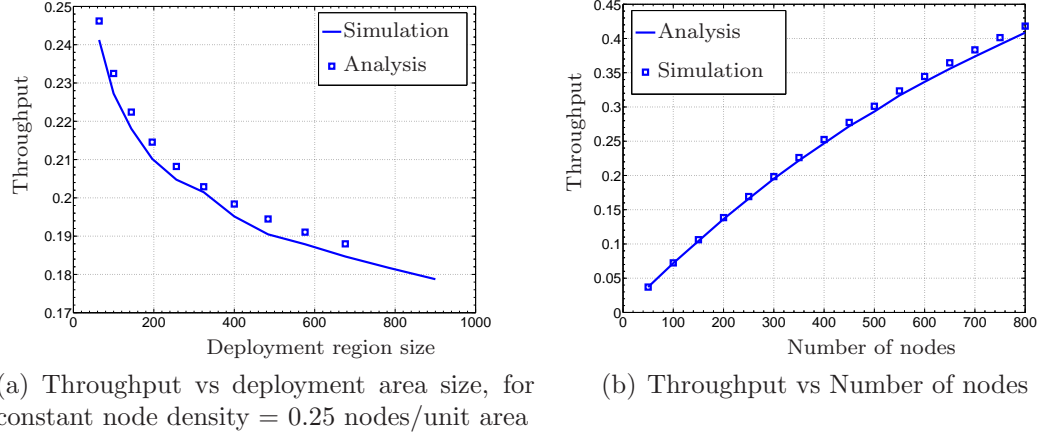


Figure 41: Simulation results for the restricted random-waypoint mobility model

5.5 Summary

This chapter presented an analysis framework for the throughput performance of SPMANETs which is versatile under various mobility scenarios. The analysis was presented for a two-hop unicast scenario. Practical considerations such as random contention, finite storage buffer, link losses, and random contact durations were seamlessly integrated into the framework by means of chain-collapsing. It was shown, as a result, that the dependence of throughput on the mobility model is captured by just three of the latter's parameters. The simulation results over various mobility models confirmed the accuracy of our analysis. Though a specific routing protocol was

used, our analysis captured the fundamental tools involved for similar analysis with different protocols. The framework can now be extended further with the same level of practicality, including for cases such as multi-hop-routing and multiple parallel unicast, as we shall see in the following chapters.

CHAPTER VI

ANALYSIS OF MULTIPLE-UNICAST STRATEGIES

This chapter seeks to employ the analytical framework developed in the previous chapter to the case when multiple unicast flows occur simultaneously in an SPMANET. In particular, we consider the case when m unique source-destination pairs attempt to communicate using n mobile relays as the medium, again employing the store, carry, and forward paradigm of SPMANETs (see Fig. 42(a)). A critical question that arises out of this is how we are to manage the limited node-memory resource so that the same can be shared by all the m different flows in a manner that will enhance the overall throughput.

Naturally, two extreme cases of buffer-management arise. These are (1) Fixed-buffer-segment (FBS) policy: Given that the buffer-size of each relay node is B , a certain chunk of the resource, $\frac{B}{m}$ in size, is dedicated to each flow. There is no mixing-up of packets belonging to different flows. In other words, only packets belonging to flow ID j is processed in the particular chunk allocated to that flow. (2) Floating-buffer (FB) policy: Packets belonging to all flows are pooled together sequentially in the buffer of every relay node. Hence, there is no prior allocation given for any given flow. These two policies are illustrated in Fig. 42(b).

The aim of this chapter is to compare the performance of the above two schemes, in order to identify the most efficient approach. As it will be shown in the following sections, the floating-buffer scheme generally achieves higher throughput than the fixed-buffer-segment scheme. Intuitively, this is due to the fact that, in the fixed-buffer-segments case, whenever the portion of the buffer resource allocated to flow i is full, it is highly unlikely that the rest of the total available buffer space is also full.

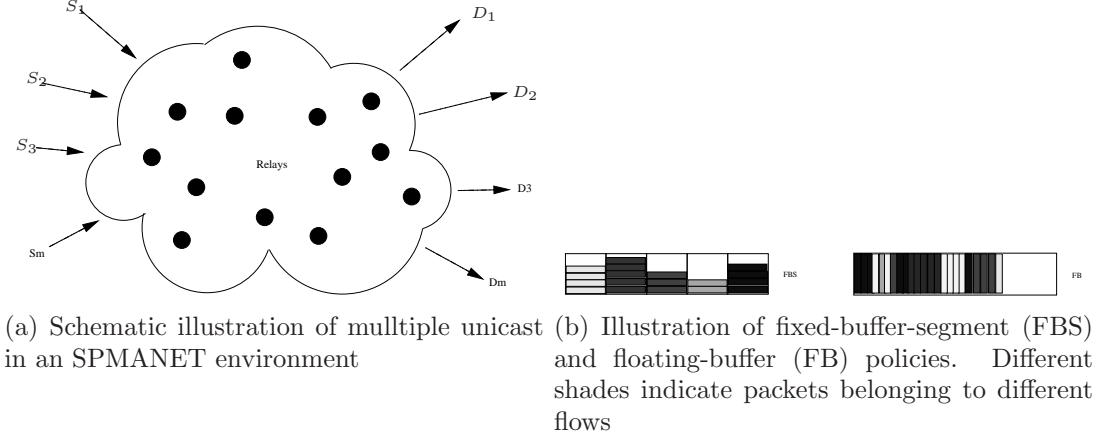


Figure 42: Network model for multiple-unicast analysis

However, the fixed-buffer-segment scheme does not make use of the same. Quantifying the resulting gain in the throughput in the case of the floating-buffer policy, we will show that the throughput achieved by the same is approximately the same as that achieved by the fixed-buffer-segments policy for a network where the relay nodes of have a buffer-size of B' each, such that $B \leq B' \leq 2B$. The exact value of B' relative to B depends upon the mobility model employed. The results pertaining to this chapter are available in our papers [73,81].

The rest of the chapter is organized as follows. In Section 6.1, we detail the notations and changes to the network model previously employed. In Section 6.2, we detail the analytical results developed. In Section 6.3, we provide a comparison of our analytical results and simulations. We then summarize our findings in Section 6.4.

6.1 Notations and Assumptions

We consider a network of n relay nodes in addition to m unique source-destination pairs. The mobility model considered in this chapter are generic, and follow the notations developed in the previous chapter. In order to simplify our notations, we do not allow the source/destination node belonging to a particular flow to act as a relay for another flow, though this extension follows easily from the analysis that follows. The approach for contention resolution is slightly different since we have m

flows competing for the same resources. This is detailed below.

6.1.1 Contention Resolution

In our model, contention resolution in our model occurs in two phases. The network goes through these two phases at every epoch. The first phase determines which source/destination node will communicate for that epoch. Let us say that there are n_s other source nodes and n_d destination nodes that are in contact with a particular source node S_j . Then the probability that S_j wins the first phase is given by $\frac{1}{n_s+n_d+1}$. Once the source/destination node is elected thus, the second phase starts. In this phase, the elected source/destination node tries to establish a link with one of the relay nodes in its range. The elected source (destination) node, always establishes a link with the corresponding destination (source) node, if both are within each others' communication range, and one block is transferred directly. Otherwise, the elected node establishes a link with one of the relay nodes in communication range (if any), at random with equal probability. Subject to memory constraints at the relay node, packets are sent/received using the link.

6.1.2 State-space Description and Throughput

As before, the state of any node v at time t in the network is designated by an ordered pair consisting of mobility-state and its buffer-occupancy, as $(\chi_v(t), b_v(t))$. In other words, at any epoch t , $\chi_v(t) \in \mathcal{S}_{mob}$, $0 \leq b_v(t) \leq B$ at any time t for any node v . Next, we define the state of the entire network as the $2(n+m)$ -tuple $\mathcal{Y}(t) \triangleq (\chi_1(t), \dots, \chi_n(t), b_1(t), \dots, b_n(t), \chi_{s_1}, \dots, \chi_{s_m}, \chi_{d_1}, \dots, \chi_{d_m})$. Here, χ_s and χ_d are the mobility states of the source and the destination. The transitions of the process $\mathcal{Y}(t)$ within its state-space can be determined from the mobility model and the underlying communication protocols. With the above state-space description of the network, we can again define the achievable throughput per flow as the *expected rate at which packets are transferred s to d when the network is in steady-state*. In

other words, $\mathcal{C}_{s_i, d_i} = \frac{\mathcal{N}_{s_i, d_i}(\tau)}{\tau}$ for the particular source-destination pair i .

In essence, we have to analyze the complex multi-dimensional random process $\mathcal{Y}(t)$ in order to derive the throughput for the SPMANET network model. Clearly, the full state-space description of the network, consisting of $\Theta(B^n |\mathcal{S}_{mob}|^{n+2m})$ states, is prohibitively large to work with.

6.2 *Setting up the embedded chain*

We now extend the methodologies explored in the previous chapter to analyze the multiple-unicast case. We will first analyze the floating-buffer scenario, from which the fixed-buffer scenario can be obtained as well. We are interested in the steady-state throughput performance for the network setup described in Section 5.1. It can be inferred that the state variable $\mathcal{Y}(t)$ describes a $2(n + m)$ -dimensional Markov Chain, since the random variables $\chi_v(t)$ etc. themselves correspond to independent random-walks, and changes in $\mathbf{b}(t)$ depend on the locations of the nodes, i.e., on the $\chi_v(t), \chi_{s_i}(t), \chi_{d_j}(t)$ variables. Hence, $\mathcal{Y}(t)$ has a unique steady-state probability distribution.

Clearly, the above state-space description is very cumbersome to work with. We circumvent this by the three-step process we developed above: (1) Suitably identifying subsets of states in which packets are sent/received (2) Constructing an embedded Markov-chain showing transitions within the above states, and (3) Collapsing the embedded chain further, thereby considerably reducing the size of the state-space. These steps are detailed in the rest of this section.

6.2.1 Construction of Embedded Chain

As in the single source-destination case, we analyze the network from a particular relay, say for node v' . Further, the average throughput for each source-destination pair is equal owing to symmetry, and hence consider only a specific source-destination pair, say S_a and D_a . We next construct the embedded chain as follows: Every time

the node v' establishes a link with any of the m sources/destinations, say S_j and D_j , (i.e., not only with the specific pair a) we mark it down along with the ID j of the corresponding source/destination pair, and the value of the network's state-variable at the end of communication in that link. Let us denote the resulting sequence as $\{\Xi_{\tau,v'}\}_{\tau=0}^{\infty}$. Then, we have the following:

Lemma 8 *For any $v' \in 1, 2, \dots, n$, the sequence $\{\Xi_{\tau,v'}\}_{\tau=0}^{\infty}$ constructs an embedded Markov-chain for the network.*

Proof: Since $\Xi_{\tau,v'}$ contains entire information about the state of the network, we can run the process Ψ from that state until a link occurs for node v' again. Thus, the probability distribution of the next instance, $\Xi_{\tau+1,v'}$ depends only on $\Xi_{\tau,v'}$.

6.2.2 Enumeration of Subsets

For the embedded chain $\Xi_{\tau,v'}$, we consider the following subsets. In each of these subsets, the subscript denotes the buffer occupancy of node v' at the end of communication with either S_a or D_a , as the case may be:

S-Group: The subsets in this group are $\mathcal{S}_{b_1,b_2,\dots,b_m}^{(a)}$ such that $0 < b_1 + b_2 + \dots + b_m \leq B, b_a > 0$. In each of these subsets, node v' establishes a link with S_a , receives a packet, resulting in the indicated buffer-state $\{b_1, b_2, \dots, b_m\}$, where b_i indicates the number of packets originating from S_i and destined for D_i .

F-Group: These are ‘buffer full’ states. The subsets in this group are $\mathcal{F}_{b_1,b_2,\dots,b_m}^{(a)}$ such that $b_1 + b_2 + \dots + b_m = B, b_i \geq 0$. In each of these subsets, node v' establishes a link with S_a , but since its buffer was already full, it received no packet, without any change in the indicated buffer-state.

D-Group: The subsets in this group are $\mathcal{D}_{b_1,b_2,\dots,b_m}^{(a)}$ such that $0 \leq b_1 + b_2 + \dots + b_m < B, b_i < B$. In each of these subsets, node v' establishes a link with D_a , sends a packet, resulting in the indicated buffer-state.

E-Group: These are ‘relatively-empty buffer’ states. The subsets in this group are

$\mathcal{E}_{b_1, \dots, b_{a-1}, 0, b_{a+1}, \dots, b_m}^{(a)}$ such that $b_1 + \dots + b_{a-1} + b_{a+1} + \dots + b_m \leq B$. In each of these subsets, node v' establishes a link with D_a , but since it had no packet for D_a to begin with, it does not send any, resulting in no change in the indicated buffer-state. Note that node v' might have had packets for other destinations, so we call these 'relatively-empty buffer' states.

Clearly, the throughput offered by the relay node v' for the source-destination pair a depends only on the steady-state probability of states in D-Group subsets. We can similarly define these subsets for any source-destination pair j . In Table 3, we illustrate the enumeration of subsets for the case $m = 2$ (two source/destination pairs). The superscript (a) is avoided for clarity.

Table 3: Enumeration of subsets for $m = 2$

| S-Group | F-Group | D-Group | E-Group |
|---|-----------------------|---|-----------------------|
| $\mathcal{S}_{1,0}, \dots, \mathcal{S}_{B,0}$ | $\mathcal{F}_{B,0}$ | $\mathcal{D}_{0,0}, \dots, \mathcal{D}_{B-1,0}$ | $\mathcal{E}_{0,0}$ |
| $\mathcal{S}_{1,1}, \dots, \mathcal{S}_{B-1,1}$ | $\mathcal{F}_{B-1,1}$ | $\mathcal{D}_{0,1}, \dots, \mathcal{D}_{B-2,1}$ | $\mathcal{E}_{0,1}$ |
| \vdots | \vdots | \vdots | \vdots |
| $\mathcal{S}_{1,B-2}, \mathcal{S}_{2,B-2}$ | $\mathcal{F}_{2,B-2}$ | $\mathcal{D}_{0,B-1}, \mathcal{D}_{1,B-2}$ | $\mathcal{E}_{0,B-2}$ |
| $\mathcal{S}_{1,B-1}$ | $\mathcal{F}_{1,B-1}$ | $\mathcal{D}_{0,B-1}$ | $\mathcal{E}_{0,B-1}$ |
| | $\mathcal{F}_{0,B}$ | | $\mathcal{E}_{0,B}$ |

Then, using Lemma 3, we form a new chain from $\Xi_{\tau, v'}$ by collapsing all the subsets in S-, F-, D-, and E-groups individually, which preserves the accumulated steady-state probabilities of each subset. We notate this collapsed chain as $\Gamma_{\tau, v'}$. The computation of the subset-averaged transition probabilities for this chain will be detailed in the steady-state analysis below.

6.2.3 Steady-state Analysis

For the rest of the analysis, we choose $a = 1$ without loss of generality. Here, we sketch the methodology employed in steady-state analysis of $\Gamma_{\tau, v'}$. We first observe the following, owing to symmetry:

Lemma 9 *Let ϖ denote any permutation of an $(m - 1)$ -tuple. For the collapsed chain $\Gamma_{\tau, v'}$, the following holds true, for any of the S -, F -, D -, and E -groups:*

$$\begin{aligned}\pi\left(\mathcal{S}_{b_1, b_2, \dots, b_m}^{(1)}\right) &= \pi\left(\mathcal{S}_{b_1, \varpi(b_2, \dots, b_m)}^{(1)}\right). \\ \pi\left(\mathcal{S}_{b_1, b_2, \dots, b_m}^{(j)}\right) &= \pi\left(\mathcal{S}_{b_j, \varpi(b_1, \dots, b_{j-1}, b_{j+1}, \dots, b_m)}^{(1)}\right).\end{aligned}$$

Now, let $\alpha(m)$ be defined as the probability that node v' , on establishing a link with one of the source/destination nodes, say S_j , subsequently establishes a link with any other source/destination node before establishing a link with S_j again. Note that $\alpha(m)$ does not depend on j since all nodes have identical mobility characteristics. Then, we have:

Lemma 10 *Let β_c be the average probability that contention is lost for a particular source j and for a particular node v' . Also, let $\alpha_0(m)$ be the probability that a relay node, currently in contact with a particular source node j , contacts any of the other $(2m - 1)$ source/destination nodes at least once, before coming into contact with j again. Then, we have:*

$$\alpha(m) = \frac{\alpha_0(m)}{\frac{2m}{2m-1}\alpha_0(m)\beta_c + (1 - \beta_c)}.$$

Proof: Consider a single source node, say S_j . Also, consider the embedded chain $\Xi_{\tau, v'}$ as defined previously. We group the states of the chain into the following subsets, with respect to S_j and the relay node in question (v'):

- S_F : Most recent contact occurred with S_j , and contention was won
- S_E : Most recent contact occurred with S_j , and contention was lost
- X_F : Most recent contact occurred with some other node than S_j , and contention was won
- X_E : Most recent contact occurred with some other node than S_j , and contention was lost

Once again, one may collapse these subsets into just four states, resulting in the new Markov chain shown in Fig. 43. Here, $\overline{\alpha}_0$ indicates $1 - \alpha_0$, and so on for the other parameters. Indicated as α_0 in the figure, the quantity $\alpha_0(m)$ is the probability that a node currently in contact with S_j will have a contact with one of the other $2m - 1$ sources/destinations, before coming in contact with the former again. Also, we define $\alpha_1 = \frac{\alpha_0}{2m-1}$ in the figure. The parameter β_c is equal to the “average probability” that a node loses contention on meeting the source/destination node. Clearly, $\alpha(m)$ as defined previously, is given from this chain by the probability that the chain in Fig. 43 visits X_F at least once before coming back to S_F again, given that it was in state S_F to start with. This probability can be computed from the fundamental matrix of the ergodic markov chain in Fig. 43 (see Chapter 2 of [76] for a discussion on the definition of the fundamental matrix for an ergodic chain). Let Z be the fundamental matrix for this chain. From Chapter 2 of [76], one can obtain the required probability as

$$\alpha(m) = \frac{\pi_{X_F}}{\pi_{S_F} \{Z_{X_F X_F} - Z_{S_F X_F}\} + \pi_{X_F} \{Z_{S_F S_F} - Z_{X_F S_F}\}}$$

where π_{S_F} and π_{X_F} are the steady-state probabilities of the respective states. Doing the necessary computations, we finally have the expression given in the lemma.

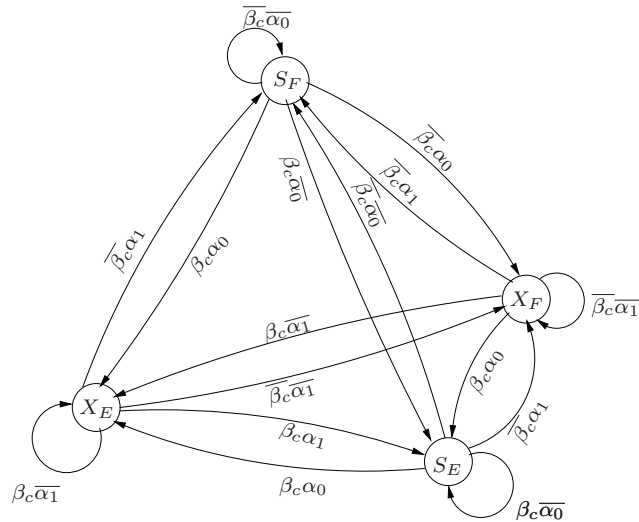


Figure 43: Embedded Markov chain approach for contention analysis

In order to complete our analysis, we need to undertake the following computations:

Computing $\alpha_0(m)$ from known mobility parameters: Let $T_0 > 0$ be the random time required for a particular contact to occur, given that the contact exists at time $t = 0$. Let $T_{\infty,m} \geq 0$ be the random time to wait until one of $(2m - 1)$ contacts occur, given that at $t = 0$ the nodes in question are distributed somewhere randomly in the grid according to the steady-state distribution of the mobility model. Let $p_{T_0}(x)$ be the density function of T_0 and $F_{T_{\infty,m}}(x)$ be the cumulative density function of $T_{\infty,m}$. Next, we note that $\alpha_0(m)$ is exactly the probability that $T_{\infty,m} \leq T_0$, which gives us—

$$\alpha_0(m) = \sum_{\tau=1}^{\infty} F_{T_{\infty,m}}(\tau) p_{T_0}(\tau). \quad (65)$$

Note that T_0 has the same distribution as in the single-unicast case, whereas $T_{\infty,m}$ has the statistics of the minimum of $(2m - 1)$ iid variables having the same distribution as T_{∞} .

Computing β_c from known parameters: To find the average contention-failure probability, β_c , we note that there are two stages in the contention phase when the network has multiple source/destination pairs. Then, $1 - \beta_c$ is the probability that contention is won by a particular source node j in the first stage, and by a particular relay node v' in the second stage. Using arguments similar to the single-unicast case, we will have the following expressions for the probability of winning contention in each stage:

$$1 - \beta_c^I = \sum_{k=0}^{2m-1} \sum_{\mathbf{x}} \frac{\pi_{spt}(\mathbf{x})}{k+1} \binom{2m-1}{k} \pi_{spt}^k(\mathbf{x}') \{1 - \pi_{spt}(\mathbf{x}')\}^{2m-1-k} \quad (66)$$

$$1 - \beta_c^{II} = \left(1 - \frac{1}{\mu_0}\right) \sum_{k=0}^{n-1} \sum_{\mathbf{x}} \frac{\pi_{spt}(\mathbf{x})}{k+1} \binom{n-1}{k} \pi_{spt}^k(\mathbf{x}') \{1 - \pi_{spt}(\mathbf{x}')\}^{n-1-k} \quad (67)$$

Then, we can compute β_c from the relation $1 - \beta_c = (1 - \beta_c^I) (1 - \beta_c^{II})$.

In general, the parameter $\alpha_0(m)$ depends on the mobility model, and β_c indicates the probability of contention failure for the potential link between a relay node with any particular source/destination node. In other words, β_c incorporates both contention between relay nodes and that between source/destination nodes in the two contention phases explained in Section 6.1.

6.2.4 Steady-State Distribution for the Collapsed Chain $\Gamma_{\tau,v'}$

Next, we proceed to complete the steady-state analysis of the collapsed chain $\Gamma_{\tau,v'}$ from which the expression for two-hop throughput will directly follow. We proceed with the analysis in the following manner. First, we note that by symmetry, any relay node will spend equal time in contacting with each of the m source/destination nodes. Hence, the steady-state probabilities of \mathcal{S}_{\dots}^j , \mathcal{D}_{\dots}^j , \mathcal{F}_{\dots}^j , and \mathcal{E}_{\dots}^j will not depend on j . Therefore, it is sufficient to determine these for $j = 1$. Next, we will compute all the steady-state probabilities corresponding to $j = 1$ in terms of one of the probabilities, namely $\pi(\mathcal{E}_{0,\dots,0}^{(1)})$. We illustrate one such step and then present the generalized lemma below. For brevity, we denote $\alpha(m)$ simply as α in the following discussion.

Continuing with the steady-state analysis of $\Gamma_{\tau,v'}$, we observe, for example, that the state $\mathcal{E}_{0,\dots,0}^{(1)}$ can be reached either from $\mathcal{E}_{0,\dots,0}^{(1)}$, or $\mathcal{D}_{0,\dots,0}^{(1)}$, or from any of the states $\mathcal{E}_{0,\dots,0}^{(j)}$, and $\mathcal{D}_{0,\dots,0}^{(j)}$ for any other j . In the first case, the relay had linked with D_1 previously, and in the second case, the relay had linked with D_j before D_1 . Using the symmetry argument from Lemma 9, we will have:

$$\begin{aligned} \pi(\mathcal{E}_{0,\dots,0}^{(1)}) &= \left\{ \pi(\mathcal{E}_{0,\dots,0}^{(1)}) + \pi(\mathcal{D}_{0,\dots,0}^{(1)}) \right\} (1 - \alpha) \\ &\quad + \left\{ \pi(\mathcal{E}_{0,\dots,0}^{(1)}) + \pi(\mathcal{D}_{0,\dots,0}^{(1)}) \right\} (m - 1) \frac{\alpha}{2m - 1}. \\ \Rightarrow \pi(\mathcal{D}_{0,\dots,0}^{(1)}) &= \frac{m\alpha}{m(1 - \alpha) + (m - 1)} \pi(\mathcal{E}_{0,\dots,0}^{(1)}) \end{aligned} \tag{68}$$

Hence, we can obtain $\pi(\mathcal{D}_{0,\dots,0}^{(1)})$ in terms of $\pi(\mathcal{E}_{0,\dots,0}^{(1)})$. Proceeding thus, using symmetry arguments, we can obtain all the steady-state probabilities for the four

groups in terms of the same.

Lemma 11 *The steady-state probabilities of all the states in the $\Gamma_{\tau,v'}$ -chain can be obtained in terms of $\pi\left(\mathcal{E}_{0,\dots,0}^{(1)}\right)$. Specifically, let $\gamma = \frac{2m\alpha}{2m-1}$. For any given l such that $m-1 \geq l \geq 1$, the following hold:*

1.

$$\pi\left(\mathcal{D}_{0,\dots,0}^{(1)}\right) = \frac{m\alpha}{m(1-\alpha) + (m-1)} \pi\left(\mathcal{E}_{0,\dots,0}^{(1)}\right) \quad (69)$$

2. *For all $(m-l)$ -tuples $(b_1, b_2, \dots, b_{m-l})$ such that each b_i is non-zero and $\sum_{i=1}^{m-l} b_i < B$, we have the following, for all k such that $0 \leq k \leq B-1 - \sum_{i=1}^{m-l} b_i$:*

$$\pi\left(\mathcal{D}_{k,b_1,\dots,b_{m-l},0,\dots,0}^{(1)}\right) = \gamma^{m-l} \pi\left(\mathcal{D}_{0,0,\dots,0}^{(1)}\right) \quad (70)$$

$$\pi\left(\mathcal{S}_{k+1,b_1,\dots,b_{m-l},0,\dots,0}^{(1)}\right) = \gamma^{m-l} \pi\left(\mathcal{D}_{0,0,\dots,0}^{(1)}\right) \quad (71)$$

3. *For all $(m-l)$ -tuples $(b'_1, b'_2, \dots, b'_{m-l})$ such that each b_i is non-zero and $\sum_{i=1}^{m-l} b_i < B$, we have the following:*

$$\pi\left(\mathcal{E}_{0,b'_1,\dots,b'_{m-l},0,\dots,0}^{(1)}\right) = \gamma^{m-l} \pi\left(\mathcal{E}_{0,0,\dots,0}^{(1)}\right) \quad (72)$$

$$\pi\left(\mathcal{F}_{(B-\sum b'_i),b'_1,\dots,b'_{m-l},0,\dots,0}^{(1)}\right) = \gamma^{m-l} \pi\left(\mathcal{E}_{0,0,\dots,0}^{(1)}\right) \quad (73)$$

4. *For all $(m-l)$ -tuples $(b'_1, b'_2, \dots, b'_{m-l})$ such that each b_i is non-zero and $\sum_{i=1}^{m-l} b_i = B$, we have the following:*

$$\pi\left(\mathcal{E}_{0,b'_1,\dots,b'_{m-l},0,\dots,0}^{(1)}\right) = \gamma^{m-l-2} \pi\left(\mathcal{D}_{0,0,\dots,0}^{(1)}\right) \quad (74)$$

$$\pi\left(\mathcal{F}_{0,b'_1,\dots,b'_{m-l},0,\dots,0}^{(1)}\right) = \gamma^{m-l-2} \pi\left(\mathcal{D}_{0,0,\dots,0}^{(1)}\right) \quad (75)$$

The rest of the analysis involves combinatorial counting of particular sets of states. From (70)-(73), it is clear that the steady-state probabilities of particular states depend upon the number of non-zero entries in the buffer occupancies corresponding to the other $(m-1)$ pairs. Hence, all we need is to count how many states are possible

with k non-zero entries, for k running from 1 to $m - 1$. Specifically, the throughput computation follows directly from the knowledge of the steady-state probabilities of the \mathcal{D} and \mathcal{E} states.

The counting of the \mathcal{D} with k non-zero buffer entries for the other $(m - 1)$ pairs will be clear by enumerating the possible subscripts: For all k -tuples of (b_1, \dots, b_k) such that $\sum_{i=1}^k b_i = j$, the possible subscripts for the \mathcal{D} -states are: $(0, b_1, \dots, b_k, 0, \dots, 0)$, $(1, b_1, \dots, b_k, 0, \dots, 0); \dots$, and $(B - j - 1, b_1, \dots, b_k, 0, \dots, 0)$. Hence, the number of states with the subscripts b_1, \dots, b_i adding to j is given by $(B - j)$ times the number of ways to divide the positive integer j into k positive integral segments. The latter is exactly given by $\binom{j-1}{k-1}$. Hence, the number of possible \mathcal{D} -states, in which a particular choice of k of the subscripts corresponding to the other $(m - 1)$ states alone being non-zero, is given by $\sum_{j=k}^{B-1} \binom{j-1}{k-1} (B - j)$. The number of ways to choose k of the $(m - 1)$ subscripts is exactly $\binom{m-1}{k}$. Hence, the number of possible \mathcal{D} -states with $k > 0$ non-zero buffer entries is given by the following, making use of the identity $\sum_{j=k}^{B-1} \binom{j}{k} = \binom{B}{k+1}$:

$$\begin{aligned} \binom{m-1}{k} \sum_{j=k}^{B-1} \binom{j-1}{k-1} (B - j) &= \binom{m-1}{k} \left\{ B \sum_{j=k}^{B-1} \binom{j-1}{k-1} - k \sum_{j=k}^{B-1} \binom{j}{k} \right\} \\ &= \binom{m-1}{k} \left\{ B \binom{B-1}{k} - k \binom{B}{k+1} \right\} = \frac{B}{k+1} \binom{m-1}{k} \binom{B-1}{k} \end{aligned}$$

Notice that from (70) each of the above states has a steady-state probability of $\gamma^k \pi \left(\mathcal{D}_{0,0,\dots,0}^{(1)} \right)$. For $k = 0$, the number of \mathcal{D} -states is clearly $= B$. Let $p = \pi \left(\mathcal{D}_{0,0,\dots,0}^{(1)} \right)$. Then, the total steady-state probabilities of all the \mathcal{D} -states is given by:

$$\pi(\mathcal{D}) = p \times B \left[1 + \sum_{k=1}^{m-1} \frac{1}{k+1} \gamma^k \binom{m-1}{k} \binom{B-1}{k} \right] \quad (76)$$

Similarly, we can derive the total steady-state probability of all the \mathcal{E} -states as

follows:

$$\begin{aligned}\pi(\mathcal{E}) &= p \frac{m(1-\alpha) + (m-1)}{m\alpha} B \left[1 + \sum_{k=1}^{m-1} \gamma^k \binom{m-1}{k} \binom{B-1}{k} \right] \\ &+ pB \sum_{k=1}^{m-1} \gamma^{k-2} \binom{m-1}{k} \binom{B-1}{k-1}\end{aligned}\quad (77)$$

Next, we observe that the throughput achieved for all source-destination pairs is given by the sum of the steady-state probabilities of the D-group states, scaled by the frequency of meeting any source/destination. We can then scale this by the number of relay nodes to obtain the total throughput per source/destination pair, as it is summarized by the theorem below:

Theorem 6 *The throughput achieved per source-destination pair for the grid-mobility model DTN network, with n relay nodes, m unique source-destination pairs, and buffer size of B data-units each, under the floating-buffer policy is given by:*

$$\mathcal{C}_{n,m,B}^{fl} = \mu_0^{-1} + \mu_0^{-1} \frac{n(1-\beta_c)B}{B + \frac{\{1 + \sum_{k=1}^{m-1} \gamma^k \binom{m-1}{k} \binom{B-1}{k}\} \alpha'(m) + \sum_{k=1}^{m-1} \gamma^{k-2} \binom{m-1}{k} \binom{B-1}{k-1}}{1 + \sum_{k=1}^{m-1} \frac{1}{k+1} \gamma^k \binom{m-1}{k} \binom{B-1}{k}}}.$$

where $\gamma = \frac{2m\alpha(m)}{2m-1}$ and $\alpha'(m) = \frac{m(1-\alpha(m)) + (m-1)}{m\alpha(m)}$.

The analysis of fixed-buffer-segment policy is a special case of the above. In this case, the buffer segments of each of the m streams do not interfere with that of any other. Hence, the throughput is the one obtained for a single source-destination pair with an available buffer of B/m . Hence, substituting B/m for B in Theorem 5, we have:

Theorem 7 *The throughput achieved per source-destination pair for the grid-mobility model DTN network, with n relay nodes, m unique source-destination pairs, and buffer size of B data-units each, under the fixed-buffer-segment policy is given by:*

$$\mathcal{C}_{n,m,B}^{fix} = \mu_0^{-1} + \mu_0^{-1} \frac{n(1-\beta_c)B}{B + m(2\beta_c - 1) + m(\alpha_0^{-1})(1-\beta_c)}.$$

6.2.5 Discussion of Analytical Results

A key outcome of the above analysis is the comparison of the performance of fixed- and floating- buffer strategies. Clearly, the fixed-buffer protocol is limited in the following manner: it does not allow packets from a particular stream to occupy the partitions provided for the other streams, even though there may be enough space in those partitions. Moreover, it is highly unlikely that all the partitions are full at the same time. Hence, the fixed-buffer strategy does not utilize the buffer resource optimally. However, the floating-buffer protocol is completely fluid and makes use of any available buffer space for any flow. In order to analytically understand how this results in a clear improvement in throughput, we take a look at the results of Theorems 7 and 6. We see that having finite buffers severely reduces the throughput, due to large term $m(\alpha_0^{-1})(1 - \beta_c)$ in Theorem 7. Although Theorem 6 has a similar term, it is considerably less, given the same set of parameters. For example, let us take a look at the case when $\gamma(B - 1) \ll 1$, which is the case when sparseness is high, and the memory availability is relatively low. Also, let us ignore the effect of contention for now (i.e., $\alpha(m) = \alpha_0(m)$). Under such a case, consider the term in the denominator of Theorem 6, which can be written as:

$$\begin{aligned}
 f &= \frac{f_1}{f_3} + \frac{f_2}{f_3}, \text{ where} \\
 f_1 &= \left\{ 1 + \sum_{k=1}^{m-1} \gamma^k \binom{m-1}{k} \binom{B-1}{k} \right\} \alpha'(m) \\
 f_2 &= \sum_{k=1}^{m-1} \gamma^{k-2} \binom{m-1}{k} \binom{B-1}{k-1} \\
 f_3 &= 1 + \sum_{k=1}^{m-1} \frac{1}{k+1} \gamma^k \binom{m-1}{k} \binom{B-1}{k}
 \end{aligned} \tag{78}$$

Let H be the largest term in the series $\sum_{k=0}^{m-1} \gamma^{k-2} \binom{m-1}{k} \binom{B-1}{k-1}$. Clearly,

$$\begin{aligned} \frac{f_1}{f_3} &= \frac{\left\{ \sum_{k=0}^{m-1} \gamma^k \binom{m-1}{k} \binom{B-1}{k} \right\} \alpha'(m)}{\sum_{k=0}^{m-1} \frac{1}{k+1} \gamma^k \binom{m-1}{k} \binom{B-1}{k}} \leq \alpha'(m) \frac{\sum_{k=0}^{m-1} H}{\sum_{k=0}^{m-1} \frac{H}{k+1}} \\ &= \alpha'(m) m \left(\sum_{k=0}^{m-1} k^{-1} \right)^{-1} \approx \frac{2}{\alpha(m) \log m} \end{aligned} \quad (79)$$

Now, employing the assumption that $\gamma(B-1) \ll 1$, we can approximate $\frac{f_2}{f_3}$ by their respective dominating terms, we have:

$$\frac{f_2}{f_3} \approx \frac{m-1}{\gamma} = \frac{m}{\alpha(m)} \quad (80)$$

Note that with increase in m , the term $\frac{f_2}{f_3}$ grows faster than $\frac{f_1}{f_3}$. Hence, f grows as fast as $\frac{f_2}{f_3}$, in other words as $\frac{m}{\alpha(m)}$. The corresponding term for the fixed-buffer policy can be written simply as $\frac{m}{\alpha(1)}$. Next, we need to understand how $\alpha(m)$ grows with m and how it compares with $\alpha(1)$. It can be shown that $\alpha(m)$ is close to $\frac{2m-1}{m} \alpha(1)$. For ease of understanding, $\alpha(m)$ is plotted against m in Fig. 44, for the random-walk-on-grid mobility model, on a 25×25 square grid, with a node communication range of 3 grid points. Hence, the denominator term in Theorem 6 is smaller than the corresponding term in Theorem 7 by about $\frac{2m-1}{m}$, under the assumptions cited previously. In general, one can expect the floating-buffer policy to be more efficient.

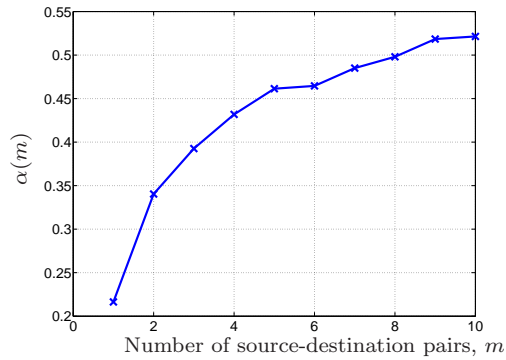


Figure 44: Variation of $\alpha(m)$ with m

6.3 *Simulation Results*

As discussed previously, the parameters α_0 , β_c , and μ_0 are hard to obtain analytically for a given mobility model. Hence, we obtained them numerically, which can be plugged into our formulations. We compare these results along with the results of simulation, as described in the following sub-sections.

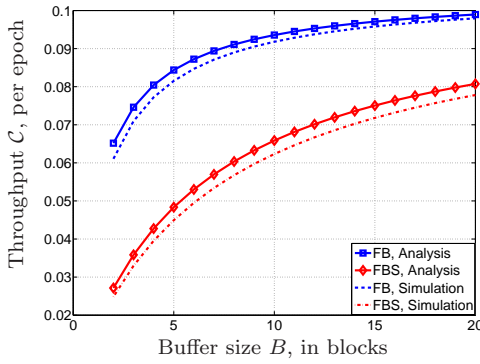
6.3.1 **Random-Walk Mobility**

We simulated a network consisting of 100 nodes on a 25×25 grid in real time with actual packet transmissions in MATLAB. We implemented both the fixed-buffer-segment (FBS) and floating-buffer (FB) policies. The communication range of each nodes was set to be $R = 2$ cells away. The block size was fixed at 200 packets (per epoch). The simulations were ran for 5 million epochs, and the results were plotted in Fig. 45. In the first experiment, $m = 5$ source-destination pairs were used in the network, varying the buffer size B per node from 2 to 16 blocks. In the second experiment, a constant buffer size of $B = 12$ blocks was used, while the number of source-destination pairs m was varied from 1 to 10.

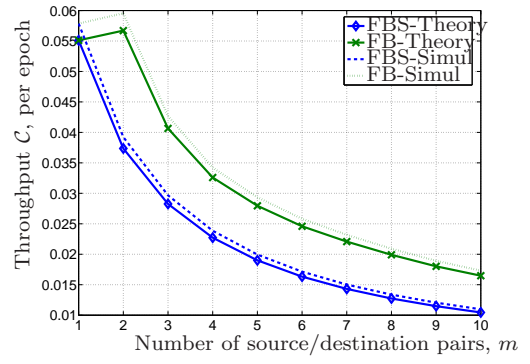
From the plots in Fig. 45, we observe that the simulation results are matched very well by theory. The first plot shows the variation of throughput with the buffer size. We see that the floating-buffer policy outperforms the fixed-buffer policy uniformly. However, both schemes approach the same saturation-value of $\mu_0^{-1} + \mu_0^{-1}n(1 - \beta_c)$. However, the critical buffer-size for a given target throughput required by the floating-buffer policy is far lower than that required by the fixed-buffer-segment policy. The second plot shows that the fixed-buffer-segment policy degrades heavily with increase in the number of source-destination pairs, whereas the degradation is lower in the case of the floating-buffer policy.

An interesting outcome of the floating-buffer policy is shown in Fig. 45(c). Here, a network of 30 relay nodes was simulated in a 25×25 grid, where the communication

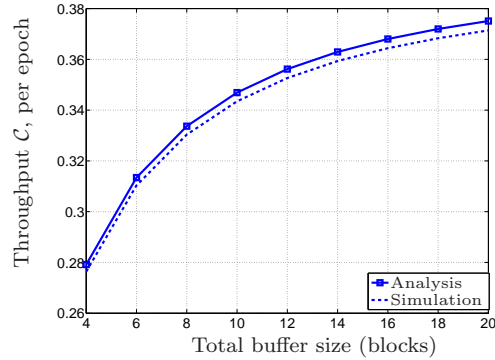
range R was chosen to be 5 grid points. The variation of throughput was studied as a function of the number of source/destination pairs while keeping the buffer availability per flow i.e., B/m constant. Note that the floating-buffer scheme provides higher throughput per source-destination pair as we increase m . However, the fixed-buffer-segment policy would provide a constant throughput per pair. This is seen easily from the analysis, and is not shown in the figure for clarity of presentation. Hence, this is an added advantage when one opts for the floating-buffer policy.



(a) Throughput vs Buffer size



(b) Throughput vs Number of source/destination pairs



(c) Throughput vs B for fixed B/m in the FB scheme

Figure 45: Simulation results for fixed- and floating-buffer multiple unicast under random walk mobility. Key: FBS = Fixed-buffer-segment; FB = Floating buffer

6.3.2 Restricted-Random-Waypoint Mobility

We use the restricted version of the random-waypoint mobility model introduced in the previous chapter to compare both simulation and analytical results. In this

mobility model, nodes choose their waypoints within the same sub-area they are currently in, with a high probability. Occasionally, it may choose a waypoint in some other area and move on. The deployment area was a $5 \text{ km} \times 5 \text{ km}$ square region wherein 100 nodes were deployed. The node velocity was chosen from a uniform random distribution, with $v_{min} = 3 \text{ m/s}$ and $v_{max} = 10 \text{ m/s}$. The communication range was chosen to be 250 m . The pause-time was modeled as an exponential distribution with a mean of 60 s . The probability of transition to a different area was 0.05. Under the multiple-unicast case, for $m = 5$, we provide in Fig. 46 the variation of throughput with the buffer size. The variation of throughput as a function of coverage area is similar in trend to the variation with grid-size under the random-walk-on-grid model. These results are avoided due to their repetitive nature.

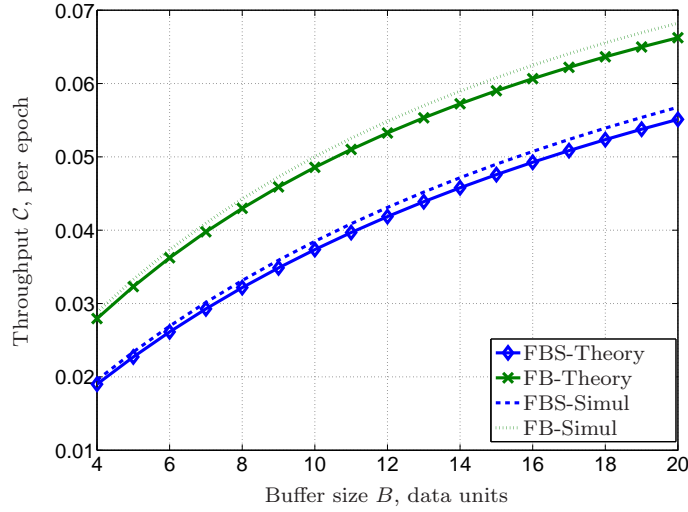


Figure 46: Multiple unicast simulation for restricted-random-waypoint mobility

The following observations can be made from our analysis and simulation results: With increase in the number of relay nodes, the throughput per source-destination pair increases sub-linearly due to the effect of contention. Similarly, the throughput per source-destination pair saturates due to contention between source/destination pairs for the limited storage space. Additionally, the effect of finite buffers can be

understood from the above formulation: for low buffer sizes, the throughput is drastically low. However, increasing the memory availability at nodes beyond a certain quantity does not give additional benefit since throughput also reaches saturation with increasing buffer-sizes. Further, the floating-buffer strategy performs better than the fixed-buffer-segment strategy universally.

6.4 *Summary*

In this chapter, we extended the generalized analysis framework developed for SP-MANETs to the case when multiple unicast flows compete for the network resources. We developed analytical estimates of the performance of two different strategies for buffer management under the multiple unicast scenario and obtained accurate estimates of steady-state throughput using the framework developed. We found that whenever multiple flows are to be accommodated, sharing the buffer-resource available in the relays by allocating different portions of the same to each flow is detrimental to the throughput. Thus, the floating-buffer policy is much more preferable over the fixed-buffer-segment way of resource-sharing.

CHAPTER VII

PERFORMANCE OF MULTIHOP ROUTING WITH BACK-PRESSURE POLICY

In this chapter, we seek to extend to apply the general Markov-chain-based framework introduced in the preceding chapters to understand the effect of multihop routing. Thus far, we have employed two-hop routing as a case study for the generalized framework. In the context of SPMANETs and DTNs, two-hop routing is usually preferred over multihop routing. The usual justification provided in favor of this choice is that multihop routing leads to an increase in the transmission cost, since intermediate nodes need to exchange packets, in addition to receiving and sending packets from the source and to the destination, respectively. Moreover, since nodes are identical, it is immaterial as to which node carries the packet to the destination. However, such considerations ignore finite-buffer considerations.

We seek to show that employing multihop routing not only gives an appreciable mitigation of the lowering of the throughput due to finite-buffer effects, but also that this gain can be achieved without having the transmission cost grow exponentially. Further, we show that a judicious policy based on back-pressure for relay-to-relay interaction achieves both with very little communication overhead. We show this by way of extending the Markov-chain-based framework, constructing a low-complexity embedded Markov chain which is then solved iteratively. In particular, we use the framework to analyze the improvement in the throughput and the energy-cost overhead incurred as a result of employing a multihop routing protocol, compared to simple two-hop relaying. Once again, we present a generalized analysis framework valid for any stochastic mobility model that exhibits statistical time-stationarity. It

was observed that throughput reduction in the finite-buffer case occurs mainly due to the high steady-state value of certain “undesirable” states in the chain. In other words, while the two-hop protocol suffers from low throughput due to finite storage, it is less severe under multihop routing since the probability of full- and empty- buffer states in the latter are considerably reduced. The results pertaining to this chapter are available in our paper [82].

The rest of the chapter is organized as follows. In Section 7.1, we present the back-pressure routing policy and the contention-resolution model used for multihop analysis. In Section 7.2, we present a sketch of the analysis developed, along with the final result leading to the iterative technique of evaluating the throughput performance. In Section 7.3, we compare the results of our analysis with the simulation results and present a short discussion on the same. We then summarize our findings in Section 7.4.

7.1 *Network Model*

We use the same models and notations for the network, mobility, and link models as in the previous chapters. In the case of multi-hop routing, the contention resolution protocol as well as the routing protocol need to be modified to suit the case. They are detailed here below.

7.1.1 Back-pressure-based Multihop Routing

Under the multihop protocol, the relay node behaves in the same manner as in the two-hop protocol, with the exception that relay-to-relay communication is also exploited in addition to the direct source-to-relay and relay-to-destination cases. Two relay-nodes while establishing contact exchange their buffer-occupancy states (number of blocks held by the buffer). If the difference in the buffer occupancies is more than or equal to 2 blocks, the relay with the higher occupancy transmits one block of packets to the relay with the lower occupancy. This constitutes a simple case of “back-pressure”

buffer-management policy.

7.1.2 Contention Resolution

When several nodes within communication range of each other try to communicate, contention between them is resolved as follows. If the source node and destination node are in direct contact, a link is always established irrespective of the presence of other relay nodes. Otherwise, if several relay nodes and a source (destination) node are in contact, any relay node is equally likely to establish a link with the source (destination) node. Finally, if several relay nodes are in contact and the source (destination) nodes are not reachable from any of the relays, any pair of relay nodes is equally likely to establish a link. In this contention model, any node within the communication range of either node that is involved in a link is not allowed to send/receive packets.

7.2 Analysis of Multihop Routing

We now analyze the performance of the multihop routing protocol defined in Section 7.1 for our SPMANET model. In order to construct a suitable embedded Markov chain, we consider the following groups of states from the full-state Markov-chain description of the network $\mathcal{Y}(t)$ defined in the same manner as in the two-hop case (cf. Section 5.2). Let us consider one particular relay node v for this discussion:

- *S-group and D-group subsets:* S_l , with $1 \leq l \leq B$ is the set of possible network states wherein the *most recent* link that node v was involved in was with the source, resulting in l blocks of q packets in the buffer after communicating with the latter. Similarly, define $D_{l'}$, for all $B - 1 \geq l' \geq 0$ to account for communication with the destination.
- *E-group and F-group states:* F is the set of possible network states wherein the most recent link that node v was involved in was with the source, but v was

unable to communicate with the latter due to lack of enough buffer space (i.e., Full/saturated buffer condition). Similarly, define E to account for the case when a link is established with the destination when the buffer-occupancy of v is zero.

- *R-group states:* R_{l_1, l_2} is the set of states such that the last contact that node v had was with a relay node v' . Moreover, v' and v had l_1 and l_2 blocks respectively in their buffers before they exchanged any packets.

We can now define an embedded Markov chain from the perspective of node v based on these subsets. In the above enumeration, note that there are B subsets each in the S - and D - groups, and $(B + 1)^2$ subsets in the R -group. Additionally, we have two other subsets: E and F . Hence, there is a total of $(B + 1)^2 + 2(B + 1)$ subsets. Hence, by using Lemma 3 we can obtain an equivalent “collapsed chain” that has just as many states. This is a huge reduction compared to the state-space of the original chain $\mathcal{Y}(t)$ that had $\Theta(B^n |\mathcal{S}_{mob}|^{n+2})$ states. Let us call this new collapsed chain Γ_v -chain. Our aim is to find an efficient way to compute the steady-state probability distribution of the states in Γ_v (which we denote by the vector $\boldsymbol{\pi}$) so that one can avoid the cumbersome simulations involving several nodes.

We note that Γ_v consists $(B + 1)^2$ extra states corresponding to relay-to-relay packet exchanges in addition to the states in the embedded chain corresponding to the two-hop relay protocol discussed in the previous chapter. The structure of Γ_v after the inclusion of R -states is shown in Fig. 47(a), where the states that are common with the corresponding two-hop chain are shown in solid bubbles. Figure 47(b) illustrates the possible states from which node v possessing 3 block can come into contact with another relay possessing 1 block (before receiving packets from v) for the case when $B = 4$. The entire chain needs to be constructed thus for any given B . The task that remains to be done is the computation of the transition probabilities, in terms of the networking and mobility parameters, between various states (i.e., p_1, p_2, p_3 etc. in

Fig. 47(b)), which we discuss in the following subsection.

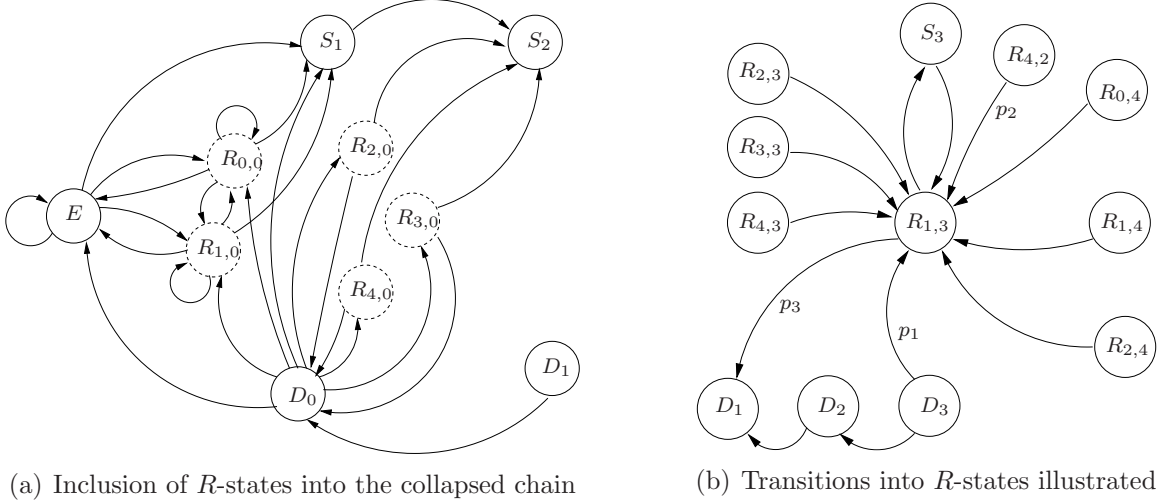


Figure 47: Composition of the Γ_v -chain illustrated for $B = 4$.

7.2.1 Computation of Transition Probabilities and Steady-State Distributions for the Γ_v -Chain

We will compute the individual transition probabilities between various states in the chain in terms of a single quantity $\alpha(n)$. We define it in the following manner: given that a particular relay node u is currently has a link with a source/destination/relay-node u' , $1 - \alpha(n)$ is the probability that u will establish a link with u' again before establishing one with any other node in the network. Next we note that since the node mobilities are not correlated, the probability that the node u will link again with a some other node v' different from u' is exactly given by $\frac{\alpha(n)}{n}$ for each v' . In other words, since there are $n + 2$ nodes, including the source and the destination, there are n nodes other than u and u' themselves and each one of them is equally likely to win the next link for node u . We will later proceed to find that $\alpha(n)$ itself is a function of (i) Mobility parameters, (ii) Networking parameters, and (iii) Contention protocol, and its computation of the same in terms of these parameters will follow.

Next, we define the following probabilities for the Γ_v -chain:

- Given that v currently has a link with the source, the probability that its next

link will be with the source again is given by p_{ss} , and the probability that its next link will be with the destination node is given by p_{sd} . Similarly, one can define p_{dd} and p_{ds} in the same manner.

- Given that v currently has a link with some relay node, the probability that the next link will be with the same/another relay node is given by p_{rr} . Similarly, we can define p_{sr} , p_{dr} , p_{rs} , and p_{rd} .

We now proceed to examine individual transition probabilities in Γ_v . Let us define Ω as the entire set of $2(B+1)$ S , D , E , and F -states. For any two states X and X' such that $X \in \Gamma_v$ and $X' \in \Omega$ such that it is possible to reach the latter directly from the former, the probability of such a transition does not depend on the current buffer-occupancy of the relay node v .¹ Hence, these probabilities will be the same as p_{ss} , p_{sd} , p_{dd} , p_{ds} , p_{rd} , or p_{rs} , as the case may be.

Due to the symmetry of our network model and due to independent mobility, one can easily verify that these probabilities are given in terms of $\alpha(n)$ by the following equations:

$$\begin{aligned} p_{ss} = p_{dd} = 1 - \alpha(n) & \quad ; \quad p_{rr} = p_{ss} + \frac{n-2}{n}(1 - p_{ss}) \\ p_{sd} = p_{ds} = p_{rd} = p_{rs} = \frac{\alpha(n)}{n} & \quad ; \quad p_{sr} = p_{dr} = \frac{n-1}{n}(1 - p_{ss}). \end{aligned}$$

The transitions into the R -states are slightly more complicated. We discuss these transitions in the proof of the following theorem, summarizing the construction of the entire embedded Markov Chain:

Theorem 8 *The steady-state distribution of the states in the collapsed embedded Chain Γ_v is given by the following system of equations. For the sake of convenience,*

¹This is due to the fact that neither mobility nor the contention protocol depends on B . The case in which the contention protocol takes into consideration buffer occupancies of the relay nodes is avoided for clarity, but is similarly possible with minor modifications to the analysis.

we define $\pi(R_{l_1, l_2})$ to be zero if either l_1 or l_2 lies outside the interval $[0, B]$ in these equations:

$$\pi(E) = p_{dd} \{\pi(E) + \pi(D_0)\} + p_{rd} \{\pi(R_{0,0}) + \pi(R_{1,0})\} \quad (81)$$

$$\pi(F) = p_{ss} \{\pi(F) + \pi(S_B)\} + p_{rs} \{\pi(R_{B,B}) + \pi(R_{B-1,B})\} \quad (82)$$

$$\pi(D_k) = \begin{cases} p_{dd}\pi(D_{k+1}) + p_{sd}\pi(S_{k+1}) \\ + p_{rd} \left\{ \sum_{j=0}^k \pi(R_{j,k+2}) + \sum_{j'=k}^{k+2} \pi(R_{j',k+1}) \right. \\ \left. + \sum_{j''=k+2}^B \pi(R_{j'',k}) \right\}, 0 \leq k \leq B-2 \\ p_{sd} \{\pi(F) + \pi(S_B)\} + p_{rd} \{\pi(R_{B-1,B}) + \pi(R_{B,B})\}, k = B-1 \end{cases} \quad (83)$$

$$\pi(S_k) = \begin{cases} p_{ds} \{\pi(E) + \pi(D_0)\} + p_{rs} \{\pi(R_{1,0}) + \pi(R_{0,0})\}, k = 1 \\ p_{ss}\pi(S_{k-1}) + p_{ds}\pi(D_{k-1}) \\ + p_{rs} \left\{ \sum_{j=0}^{k-2} \pi(R_{j,k}) + \sum_{j'=k-2}^k \pi(R_{j',k-1}) \right. \\ \left. + \sum_{j''=k}^B \pi(R_{j'',k-2}) \right\}, 1 < k \leq B \end{cases} \quad (84)$$

For any $0 \leq l_1 \leq B$,

$$\varphi_{l_1, l_2}^{-1} \pi(R_{l_1, l_2}) = \begin{cases} p_{dr} \{\pi(E) + \pi(D_0)\} + p_{rr} \{\pi(R_{0,0}) + \pi(R_{1,0})\}, l_2 = 0 \\ p_{sr} \{\pi(F) + \pi(S_B)\} + p_{rr} \{\pi(R_{B,B}) + \pi(R_{B-1,B})\}, l_2 = B \\ p_{sr}\pi(S_{l_2}) + p_{dr}\pi(D_{l_2}) \\ + p_{rr} \left\{ \sum_{j=0}^{l_2-1} \pi(R_{j, l_2+1}) + \sum_{j'=l_2-1}^{l_2+1} \pi(R_{j', l_2}) \right. \\ \left. + \sum_{j''=l_2+1}^B \pi(R_{j'', l_2-1}) \right\}, 0 < l_2 < B \end{cases} \quad (85)$$

$$\text{where } \varphi_{l_1, l_2} \triangleq \frac{\pi(R_{l_2, l_1})}{\sum_{j=0}^B \pi(R_{j, l_1})} \text{ for any } 0 \leq l_1, l_2 \leq B.$$

Proof: The proof for the steady state equations (81)-(84) follow from the previous discussion in this section. We only need to analyze the transitions into the R -states. Given that the current link for node v is with a source, destination, or some relay, the probability that the next link is with any of the $n - 1$ other relays is given exactly by the quantity p_{rr} . Now in order to determine the probability that the relay node corresponding to the new link has exactly l_1 blocks of q packets in its buffer, we can use the chain collapsing principle. Equivalently, we only need to determine the “subset-averaged” probability distribution of the buffer occupancy of the any relay node every time v comes into contact with the same. By symmetry, we can say that this distribution, denoted as φ above is exactly the same as the distribution of node v every time it comes into contact with a relay node that has occupancy l_1 . We call φ as the “joint buffer-occupancy” distributions for the chain Γ_v . Hence, the expression for φ is exactly as given in (85). Knowing this, we can then compute the steady-state probability of each R -state by carefully examining the possible previous states for v in Γ_v .

The analysis of steady-state for the Γ_v chain would be complete once we determine the unknown parameter $\alpha(n)$ in terms of networking and mobility parameters. Doing so would enable us to study scaling laws with respect to network size (n), node-density, mobility characteristics, etc. The relevant analysis follows in the next subsection.

7.2.2 Contention Analysis and Mobility Parameters

The computation of $\alpha(n)$ in closed form necessitates us to look into the effect of node-to-node contention (which has not been considered previously). We again derive the same similar to the manner described in the previous chapter. We again need to define a four-state embedded chain as follows:

- U_W : Most recent contact occurred with U , and contention was won
- U_L : Most recent contact occurred with U , and contention was lost

- X_W : Most recent contact occurred with some other node than U , and contention was won
- X_L : Most recent contact occurred with some other node than U , and contention was lost

Once again, $\alpha(n)$ in the original chain Γ_v can be derived from the four-state contention chain as the probability that the latter, starting at state U_W , goes through the state X_W at least once before reaching U_W again. Doing the necessary computations, we have the final expression below:

In order to complete our analysis, we need to undertake the computation of $\alpha_0(n)$ and β_c in terms of known networking parameters. Again, these can be computed in a manner similar to the previous chapter. The expression for β_c remains the same, whereas the expression for $\alpha_0(n)$ can be obtained as:

$$\alpha_0(n) = p' E[T_0] \frac{1 - (1 - p')^n}{1 - (1 - p')^{n+1}} \approx \frac{E[T_0]}{E[T_\infty]} \frac{n}{n+1}. \quad (86)$$

This completes the construction of the collapsed chain Γ_v . In order to complete the analysis, one needs to determine the steady-state probabilities of all the states in this chain. This is done by solving the equations (81)-(85). However, we note that the steady-state equations for the R -states (85) are non-linear. Hence, it is impossible to obtain closed-form solutions for the throughput. Nevertheless, (81)-(85) can be solved by iterative methods. One such method that we employ is described below:

1. Assume an initial steady-state distribution $\boldsymbol{\pi}_0$ such that

$$\sum_{j_1=0}^{B-1} \pi_0(D_{j_1}) + \sum_{j_2=1}^B \pi_0(S_{j_2}) + \sum_{k=0}^B \sum_{l=0}^B \pi_0(R_{k,l}) + \pi_0(E) + \pi_0(F) = 1.$$

- . We then go into the first round of iteration.

2. During the i^{th} round of the iteration, compute the new values of the steady-state parameters $\boldsymbol{\pi}_i$ using the formulations given in (81)-(85) using $\boldsymbol{\pi}_{i-1}$ in place of the steady-state values $\boldsymbol{\pi}$.
3. Repeat step 2 until $\|(\boldsymbol{\pi}_i) - (\boldsymbol{\pi}_{i-1})\|_1 < \epsilon$, where ϵ is the choice of error tolerance.

In general, this queuing-theoretic model converges within 10-20 iterations for an error tolerance of 1×10^{-6} . In contrast, simulating the exact network for a given n takes a few million epochs for the system to reach steady-state. In addition, the entire simulation has to be repeated in order to understand scalability issues, effects of various parameters, etc.

Having obtained the steady-state distribution of Γ_v thus, we can compute the throughput (in blocks per epoch) contributed by all the n relay nodes for multihop unicast from Proposition 1 as follows:

$$\mathcal{C}_{s,d} = \frac{1}{E[T_0]} \frac{n(1 - \beta_c)}{E[T_0]} \frac{\sum_{j=0}^{B-1} \pi(D_j)}{\pi(E) + \sum_{j=0}^{B-1} \pi(D_j)}. \quad (87)$$

The above expression follows from the fact that the throughput is given by the ratio of the total steady-state probability of the “desirable” states (in Γ_v , these are the D -states) to the total steady-state probability of all states where the destination node is linked with (i.e., E - and D -states), times the frequency of establishing a link with the destination which can be computed as $\frac{(1-\beta_c)}{E[T_0]}$. The first term in the throughput indicates the contribution of direct source-to-destination contacts.

As we shall see in the discussion of results, multihop relaying offers higher throughput than the corresponding two-hop protocol under the same network setup. However, it comes at the price of higher energy spent due to additional relay-to-relay transmissions. This trade-off can also be quantified using our framework. In particular, the number of additional transmissions $\mathcal{T}_{s,d}$ incurred per block delivered can be obtained

in terms of the steady-state probabilities of Γ_v as:

$$\mathcal{T}_{s,d} = \frac{\sum_{j=0}^B \sum_{j'=0}^{j-2} \pi(R_{j,j'})}{\sum_{j=0}^{B-1} \pi(D_j)}. \quad (88)$$

since exactly one packet is transmitted by v in each of the R -states in the above expression.

7.3 *Simulation Results and Discussion*

We verified our analysis and iterative methods of obtaining the throughput and energy cost of multihop routing by means of MATLAB simulations. A general observation is that while simulation of one instance of the whole system takes anywhere from a few hours to a day, our iterative queuing-theoretic framework can obtain the entire characteristics of the network as a function of various parameters, in a few seconds. In our case, we used the random-walk-on grid and the random-waypoint mobility models described in the previous chapters.

7.3.1 Discussion of Results

In order to study the effect of finite buffers on the throughput of multihop routing vs two-hop routing, we simulated an SPMANET network consisting of 30 nodes on a 25×25 grid with a communication radius of $R = 2$ grid points. For buffer sizes ranging from $B = 2$ to $B = 16$ blocks of 100 packets each per node, we plotted the observed throughput in Fig. 48(a). In addition, the corresponding plots for the transmission cost incurred have also been plotted in Fig. 7.3.1. We have also plotted the same obtained using our iterative queuing-theoretic framework. In addition, the throughput for two-hop routing under the same conditions is also shown in the figure, using the analysis developed in the previous chapter.

Clearly, multihop routing gives a consistent gain in the throughput vis-a-vis two-hop routing. As to how such an improvement occurs, we refer to the plot in Fig. 49. Here, the steady-state buffer-occupancy distribution of relay nodes that link with

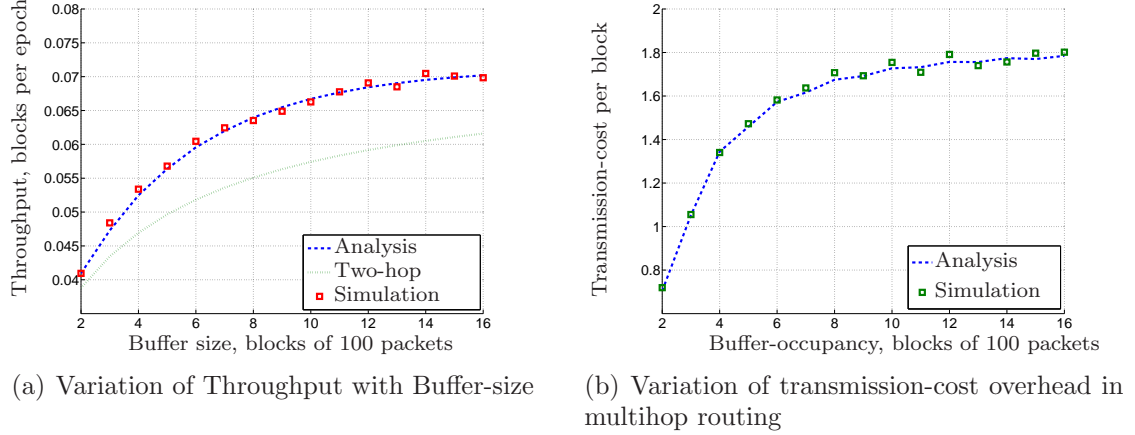


Figure 48: Simulation Results for the Random-Walk-on-Grid Mobility

the destination are shown for the case $B = 8$. Reduction in throughput for lower buffer limits, as shown in Fig. 48(a) occurs as a result of the possibility of a node coming in contact with the destination to have no packets in its buffer. However, from Fig. 49 we see that the chance of this “empty-buffer state” is considerably lower under multihop routing in contrast to two-hop routing. This, in turn is due to the back-pressure buffer management strategy described in Section 7.1 which aims to push the distribution of buffer-occupancy away from the “empty” and “full” states, so that a relay node has a higher chance of both accepting a block from the source (avoiding buffer-overflow) as well as delivering a block to the destination. In fact, the mitigation of finite-buffer effects is expected to improve for higher n since the chances of relay-to-relay contacts improve. This is confirmed by the buffer-occupancy distribution plots (during contacts with the destination node) for $n = 20$ and $n = 40$ in Fig. 50 for the same network parameters. Clearly, the distribution has higher concentration at the non-empty states for higher n .

Next, we look at the effects of contention and node-density and understand the scaling behavior of multihop routing. We do this for the restricted-random-waypoint mobility model. The set of parameters chosen for this model was the same as the one we employed for the two-hop-single-unicast case. In Fig. 51(a), the per-node

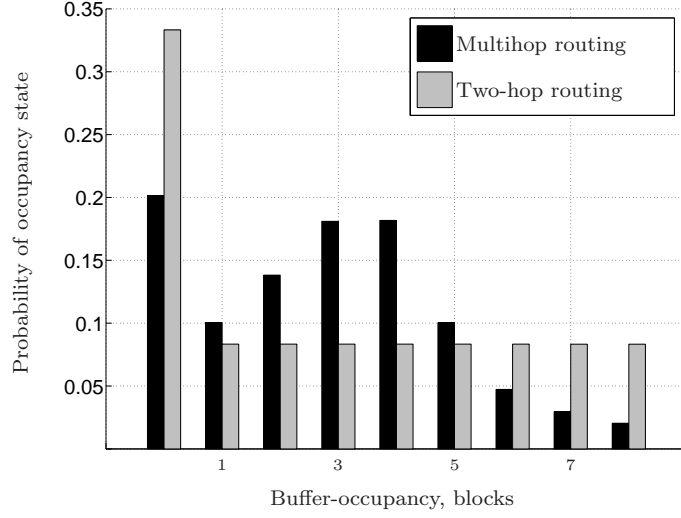


Figure 49: Steady-state buffer-occupancy distribution for $B = 8$

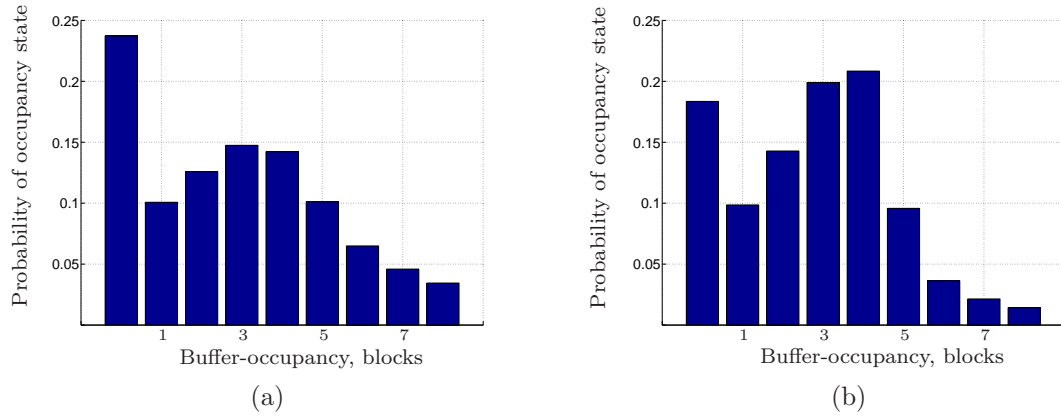
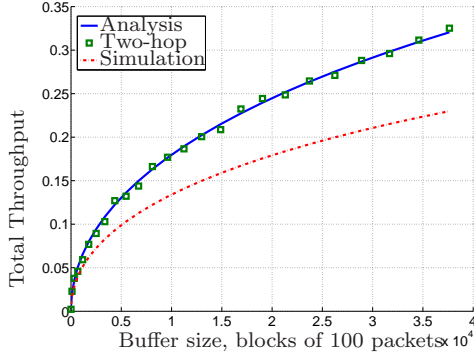


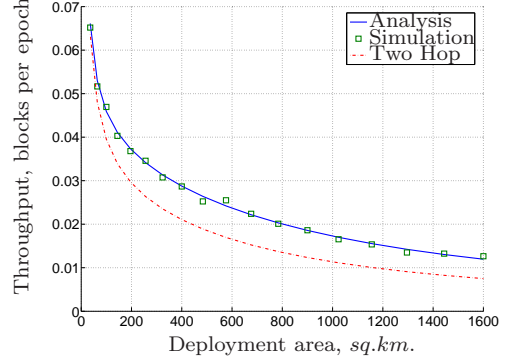
Figure 50: Steady-state buffer occupancy during link w/ destination node, for $n = 20$ and $n = 40$

throughput contribution in the network is plotted as a function of the number of nodes for a fixed area-size of 100km^2 . Clearly, the per-node throughput diminishes when n is large. This is due to the effect of node-to-node contention. For very high n , it was observed that the total throughput in the network saturates.

Generally, it is observed that with increase in contention between nodes, the buffer-occupancy distribution during links with the destination shift away from the empty state. Improvement in throughput due to this slight mitigation is however diminished considerably since the probability of establishing a link itself goes down under heavy contention, since the term $1 - \beta_c$ dominates in (87). In Fig. 7.3.1, a plot of the throughput is shown for the case when the deployment area is increased from 36 to 1600km^2 while keeping the relay-node density constant at 1 node for every 10km^2 . Clearly, the plot shows that it is not enough to scale the number of nodes linearly in proportion to the size of the area in order to obtain the same throughput.



(a) Throughput vs Number of relay nodes



(b) Throughput vs Deployment area for fixed node-density

Figure 51: Simulation Results for the Restricted-Random-Waypoint Mobility

7.4 Summary

The throughput capacity achievable in a sparse mobile network under a class of multihop single-copy unicast protocols were analyzed in this chapter. The specific protocol considered was based on simple back-pressure buffer management. Using

our framework, accurate estimates of steady-state throughput and transmission-cost overhead were derived for multi-hop routing. The effect of various network parameters on the performance of the network was thus analyzed. The accuracy of the analysis was validated by simulations of the network model in real time.

CHAPTER VIII

PERFORMANCE OF NETWORK-CODED MULTICAST

Thus far, we have focused on single-copy routing in a stochastic SPMANET model under the unicast scenario. In this chapter, we study the performance of a multicast scheme enhanced by Network Coding. We seek to analyze the performance of Multicast enabled by Network Coding and compare its performance to a simple custodial-multicast scheme. Network Coding (NC) has previously been suggested in the context of SPMANETs in works such as [83,84], its impact on multicast has hardly been studied. Again, the analysis presented in [84] does not involve performance at steady-state but involves the latency performance of an isolated burst of packets. It is noted that Karande *et al* have shown in [85,86] that Network Coding does not change the order of throughput in a stationary ad-hoc network, since the same can be achieved by simple store and forward methods employing multipoint transmission and reception. Hence, we have aimed at understanding the performance benefits of multicast with network coding (specifically RLC i.e., Random Linear Coding) in an SPMANET setting. Since multicasting without network coding has not been delved deeply in the context of SPMANETs and involve several issues to be settled [87], we use a simple custodial transfer scheme for comparison.

Even though accurate analysis of network-coded multicast is very complicated, we derive a provable way to obtain exact performance. We then build an embedded Markov chain to analyze the steady-state throughput performance of the network-coded schemes under this setup, which is then solved iteratively in an approach similar to the multihop-routing scenario of the previous chapter. Our results show that the network-coding-based scheme offers considerable improvement for the case when the

storage size of the relay nodes is small and when the number of destination nodes is large. In addition, we are interested in obtaining regimes under which network coding performs significantly better than a simple custodial scheme that implicitly replicates packets from the source as many times as the number of destinations served. We find that network coding has the dual benefit as a result of better buffer-management, and as a result of the *coupon-collector* effect [84] . However, we observe that significant improvements are achievable only when the number of destinations served is large and when the relay-node buffer sizes are not very large.

8.1 *Network Model*

We consider the generalized SPMANET model developed in the previous chapters. In brief, we consider a network of n mobile relay nodes in a certain deployment area, with identical mobility statistics derived from a model exhibiting stochastic time-stationarity. To suit the scenario of multicast, the network has a single mobile source serving m mobile destination nodes. In order to facilitate the analysis, we assume that at each epoch, a single packet can be transmitted, and that the buffer size of the relay nodes is B packets each.

We now describe the routing protocols for multicast that we compare in this chapter. It is noted that under both schemes, we need to allow for relay-to-relay communication so that we are not comparing two schemes with poor performance. In the first case, multihop routing makes efficient use of source-to-relay contacts so that replication of packets happens internally and hence the source itself requires each packet to be transmitted only once. In the second case, multihop routing helps to decrease the chance of having redundant packets by increasing diversity.

8.2 Multicast Routing Protocols

8.2.1 Network-Coded Multicast

We employ RLC coding identical to [83] in order to improve steady-state throughput under multicast. Under this scheme, packets are considered to be vectors in a finite field of desirably large size, say $GF(2^{n_c})$. Whenever a packet is to be sent by the relay to one of the m destinations (or another relay) the former creates a random linear combination of the B entries in its buffer and sends the resulting packet. Again, the coefficients are chosen from $GF(2^{n_c})$. When a relay-to-relay link occurs, one of them is randomly chosen as the sender and a linear combination of its contents are sent to the other. Here, relay-to-relay transmissions is only possible with a blind strategy since it is impossible to keep track of the relay nodes' measure of innovativeness with respect to all the other relays and destinations. Throughout this chapter, we assume that the relay nodes never need to delete the contents in its buffer. Hence, the buffers of the relay nodes are physically full for ever after reaching steady state. Even though physically full, it is clear that the buffers may have redundancies in them, and hence a particular destination might find fewer than B packets that are linearly independent from the ones already received. Also, the destinations do not need to recover the original packets, but are satisfied if they receive a packet that is linearly independent from all the packets received in the past. These scenarios are illustrated in Figs. 52, 53.

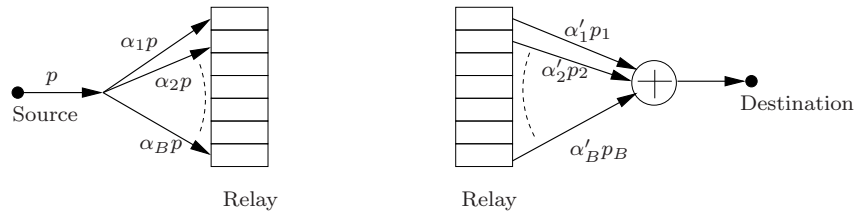


Figure 52: Network-Coded Multicast: Interaction of a relay node with the source and a destination node

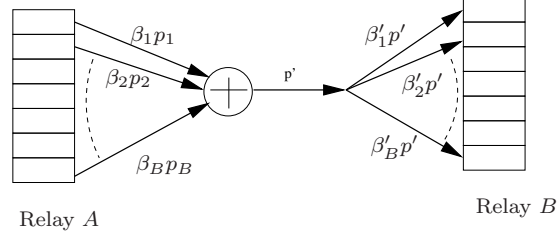


Figure 53: Network-Coded Multicast: Interaction between two relay nodes

8.2.2 Simple Custodial Multicast

The comparison scheme used in the chapter is a version of custodial transfer modified to suit multicast [87]. In this scenario, each packet is assumed to be uniquely identifiable. On linking with the source, a relay node, if it has an empty buffer space, will accept a packet and assign a counter to it, initialized as m . It is also assumed that when a relay node links with one of the destinations, the latter identifies those packets from the former's buffer that it has not yet received, so that the former sends one of them at random if available. Before the beginning of the next epoch, the relay node decreases the counter associated with the transmitted packet by 1. Packets which have a counter value of zero are discarded before the beginning of next epoch, making room for more incoming packets.

Whenever a link occurs between two relay nodes, they make each other's counter totals known to the other in addition to the IDs of the packets in their buffers. Now, a back-pressure policy is employed to determine which node will send packets. Whichever node has a total counter value higher than the other by at least two will be the sender, and the other one will be the receiver. Then, a packet is sent if the receiver has an empty space in its buffer. Let us say that the packet sent had a counter value of k in the sender's buffer. At the end of the transfer, the sender will update this counter to $\lceil \frac{k}{2} \rceil$ and the receiver will accept the packet with $\lfloor \frac{k}{2} \rfloor$ as the counter for its copy of the same packet. Additionally, if a copy of the same packet is present already in the receiver's buffer, the latter combines both and adds on the

value $\lfloor \frac{k}{2} \rfloor$ to the index in the copy. Hence, this protocol aims at redistributing the transmission load among the relay nodes.

8.3 Analysis of Network-Coded Multicast

From the description of the network-coded scheme, it is clear that obtaining the steady-state performance of this scheme for the SPMANET network model is a cumbersome task. This is because in order to find whether a received packet is innovative, the destination node needs to keep track of all the packets received in the past and has to perform Gaussian elimination on an ever-increasing rank matrix. Hence, only an approximate solution can be obtained by decoding a burst of K packets such that K is sufficiently large. However, this does not help us determine whether the performance obtained thus is an upper bound or a lower bound. Addressing these challenges, we provide the following analysis methodology making use of Markov Chain techniques. In addition, the solution for steady-state distributions can be computed in negligible time for a range of design parameters as the complexity of these chains is manageable and scalable.

8.3.1 The “Stream Separation” Argument

As described previously, throughput is defined from the standpoint of the rate at which linearly-independent packets are received at the destination. In other words, if a received packet is a linear combination of the past history, it does not contribute to the throughput. In other words, for $i \neq j$ if a packet is delivered by a relay v to destination D_i , it does not affect the number of innovative packets in v 's from the standpoint of D_j . Hence, there is no interaction among the m different streams that the node v is serving. To explain further, it is as if the relay v has m different buffers of size B each dedicated to each of the destinations. Additionally, the throughput contributed by any relay node, on the average, will be equally distributed among all

the m destinations. Hence, we can conduct our analysis from the standpoint of a particular destination node, say D_1 . This means that the analysis of multicast with RLC encoding is the same as the analysis of multihop unicast protocol with RLC encoding. In the following subsections, we will show that the throughput performance of the latter scenario, which is the same as the per-destination performance of the former, can be upper-bounded by the performance of simple relaying with backpressure-based multihop routing that we derived in the previous chapter.

8.3.2 Preliminary Steps in Analysis

Unlike previous scenarios, we cannot form a simple state-space consisting of the individual buffer-occupancies of each node. In the earlier case where no correlations exist between the buffer contents of any two given nodes, it was possible to trace the number of useful packets that a given relay can contribute to the destination independent of the buffer-states of the other relays. However, in the case of RLC-encoded relay-to-relay transfer that we have employed in this case of multicast communication, redundancies and correlations between the contents of any two relays' buffers are expected. As a simple illustration, let us take the case where relay u has 4 innovative packets not available in w in its buffer and relay w has 2 innovative packets not available in u in its buffer. Let us say u establishes a link with w and transfers a single linear combination of its buffer contents to the latter. Now, since u does not modify the contents of its buffer, it still has 4 innovative packets that it can contribute to the destination. This means that though w will now have 3 innovative packets that it can contribute to the destination node, u and w will together still have only 6 innovative packets destined for D_1 . In addition, we have a peculiar phenomenon resulting from the following scenario: Assuming that the next series of links are established between u and the destination node, and that it occurs four consecutive times, the destination node will have 4 packets delivered to it by u . However, the occupancy of w drops by

1 even though it is not involved in any link in the mean time.

From the above discussion, it is clear that we now need to trace the occupancies of not only individual nodes with respect to the destination, but also those of different subsets. Additionally, we will also need to trace the “relative” occupancies between different subsets. Since $2^n - 1$ different subsets can be constructed from a set of n different nodes, we will then have a matrix of size $2^n - 1 \times 2^n - 1$ buffer occupancies to trace. This is prohibitively large. We can simplify this scenario by means of the methodology described below.

8.3.3 Construction of the Occupancy Matrix and the Complementary Occupancy Vector

Let $[n] \triangleq \{1, 2, \dots, n\}$. Let nodes $1, \dots, n$ each have B linear combinations of packets M_1, \dots, M_l . In other words, let

$$P(i, j) = \sum_{k=1}^l a_k M_k, i \in [n], j \in [B]. \quad (89)$$

where a_k are coefficients in the chosen finite field $GF(2^{n_c})$.

Let $\mathcal{V}(S) \triangleq \text{span} \{P(i, j) | i \in S, j \in [B]\}$ for all $S \subseteq [n]$.

Definition 1 For any two subsets of relay nodes $U, W \subseteq [n]$, we define the occupancy of U w.r.t. W as:

$$\Phi(U, W) \triangleq \dim(\mathcal{V}(U)) - \dim(\mathcal{V}(U) \cap \mathcal{V}(W)).$$

We define the occupancy matrix for the entire network to be the map $\Phi : 2^{[n]} \times 2^{[n]} \rightarrow \mathbb{N}$, where Φ is defined as above.

Definition 2 The complementary occupancy vector for our network model is defined as the map $\Psi : 2^{[n]} \rightarrow \mathbb{N}$, where $\Psi(S) \triangleq \Phi(S, S^c)$ for any $S \subseteq [n]$, and $S^c \triangleq [n] \setminus S$.

We will now show that there is a one-to-one correspondence between the occupancy matrix and the complementary occupancy vector defined above, so that one can use the latter for describing the buffer-states in the network.

Lemma 12 For any two subsets of nodes $U = \{u_1, u_2, \dots, u_k\} \subseteq [n]$, and $W \subseteq [n]$. Then, $\Phi(U, W) = \sum_{j=1}^k \Phi(u_j, W \cup \{u_1, \dots, u_{j-1}\})$.

Proof: Clearly,

$$\begin{aligned}
\sum_{j=1}^k \Phi(\{u_j\}, W \cup \{u_1, \dots, u_{j-1}\}) &= \sum_{j=1}^k [\dim(\mathcal{V}(\{u_j\})) \\
&\quad - \dim(\mathcal{V}(\{u_j\}) \cap \mathcal{V}(W \cup \{u_1, \dots, u_{j-1}\}))] \\
&= \sum_{j=1}^k [\dim(\mathcal{V}(W \cup \{u_1, \dots, u_j\})) \\
&\quad - \dim(\mathcal{V}(W \cup \{u_1, \dots, u_{j-1}\}))] \\
&= \dim \mathcal{V}(U \cup W) - \dim \mathcal{V}(W) \\
&= \dim \mathcal{V}(U) - \dim(\mathcal{V}(U) \cap \mathcal{V}(W)) \\
&= \Phi(U, W). \tag{90}
\end{aligned}$$

Corollary 2 $\forall S_1, S_2, S_3 \subseteq [n], \Phi(S_1 \cup S_2, S_3) = \Phi(S_1, S_3) + \Phi(S_2, S_1 \cup S_3)$.

Corollary 3 $\forall S_1, S_2, S_3 \subseteq [n]$ such that $S_1 \subseteq S_3$, $\Phi(S_1 \cup S_2, S_3) = \Phi(S_2, S_1 \cup S_3) = \Phi(S_2, S_3)$.

Given $n \in \mathbb{N}$, we say that the vector of non-negative integers $\{b_S\}_{S \subseteq [n]}$ is a *feasible occupancy vector* if there exists a $l \in \mathbb{N}$ and a set of packets $P(i, j)$ defined as in (89) such that $\Psi(S) = \Phi(S, S^c) = b_S$. Then, the following theorem holds:

Theorem 9 For any given set of packets $\{P(i, j)\}_{i \in [n], j \in [B]}$, the set of complementary occupancy vectors and the occupancy matrices are in one-to-one correspondence.

Proof: From the statement of Lemma 12, we see that $\Phi(\{i\}, S)$ for any $i \in [n], S \subseteq [n]$ generates the entire occupancy matrix. Hence it is enough to show that a given feasible occupancy vector Ψ uniquely determines $\Phi(\{i\}, S)$ for any $i \in [n], S \subseteq [n]$. Additionally, since $\Phi(\{i\}, S) = 0$ when $i \in S$, we only need to show this for the case when $i \notin S$. Specifically,

$$\Phi(S^C \cup \{i\}, S) = \Phi(\{i\}, S) + \Phi(S^c, S \cup \{i\}) \quad (91)$$

$$\begin{aligned} \Rightarrow \Phi(\{i\}, S) &= \Phi(S^c \cup \{i\}, S) - \Phi(S^c, S \cup \{i\}) \\ &= \Phi(S^c, S) - \Phi((S \cup \{i\})^c, S \cup \{i\}) \end{aligned} \quad (92)$$

$$= \Psi(S^c) - \Psi((S \cup \{i\})^c). \quad (93)$$

where (91) follows from Corollary 2 and (92) follows from applying Corollary 3 on the expression $\Phi(\{i\} \cup S^c \setminus \{i\}, S \cup \{i\})$. This completes the proof of Theorem 9. Note that the complementary occupancy vector Ψ has length $2^n - 1$ ignoring the case $S = \emptyset$.

We now summarize a few important properties of the complementary occupancy vector that will be useful in the construction of the embedded chain for the network.

Corollary 4 (Monotonicity) *From (93), it is clear that for any $U, W \subseteq [n]$ such that $U \subseteq W$, $\Psi(U) \leq \Psi(W)$.*

Corollary 5 (Consistency) *For any $S \in [n]$, it is clear that $\Psi(S) \leq B|S|$ where $|\cdot|$ denotes cardinality. This follows from a simple observation of the definition of joint occupancy.*

Corollary 6 (Innovativeness) *For any $i \in [n]$, the number of innovative packets that i is capable of delivering to the destination is given by $\Psi([n]) - \Psi(\{i\}^c)$. This follows from the observation that $\Psi([n])$ is the total innovativeness in all the relays and that $\Psi(\{i\}^c)$ is the number of innovative packets in the network that cannot be generated by the contents of the buffer in i .*

Corollary 7 (Mutual Occupancy) *The number of innovative packets in relay i innovative to (i.e., that cannot be generated by) the contents in relay j is given by:*

$$\mu(i, j) = \Psi(\{j\}^c) - \Psi(\{i, j\}^c) \quad (94)$$

8.3.4 Construction of the Network-level Chain

As before, the state vector at time t for the entire network contains the vector of mobility states $(\chi_1(t), \dots, \chi_n(t), \chi_s(t), \chi_{d_1}(t))$. Here, $\chi_{d_1}(t)$ denotes the mobility state of the particular destination node of interest, i.e., D_1 . However, instead of the n -length occupancy vector that has been in use in previous chapters, we now need to use the $2^n - 1$ -length occupancy vector that we have defined in the preceding discussion. Hence, we define the state of the entire network now as the pair:

$$\mathcal{Y}(t) \triangleq \left[(\chi_1(t), \dots, \chi_n(t), \chi_s(t), \chi_d(t)), \{b_S\}_{S \subseteq [n]} \right]$$

We now need to discuss how the occupancy states are updated. This would complete the description of the entire chain for the network. We first define the following operation on the complementary occupancy vector:

Definition 3 (Augmentation) *“Augmenting the S -entry w.r.t. i ” of the complementary occupancy vector for any given $S \subseteq [n]$ and $i \in S$ consists of the following recursive operation:*

- (a.) If $b_S = B |S|$, stop. Otherwise: (b.) add 1 to b_S , and (c.) If $|S| = 1$, stop. Otherwise, for every $j \in S$ such that the augmentation results in $b_S - b_{S \setminus \{j\}} > B$, augment $S \setminus \{j\}$ w.r.t. i .

The vector b_S for all $S \subseteq [n]$ is said to be “ S -augmentable” if the situation (a.) never occurs during the course of the augmentation.

We can now determine how the occupancy vectors are updated for each of the following cases:

- *Packet arrival:* If node i establishes a link with the source, then augment each $\{i\} \cup S$ -entry in the order of cardinality of the S ’s.

- *Packet delivery:* If node i establishes a link with the destination, then decrease each non-zero b_S such that $i \in S$ by 1.
- *Relay-to-relay interaction:* If node i establishes a link with node j and i transfers a packet to j , then decrease every $b_{S'}$ such that $i \in S'$ and $j \notin S'$ by 1 if $b_{\{j\}^c} - b_{\{i,j\}^c}$ was non-zero to begin with.

It can be shown that the above set of operations always result in a feasible set of b_S . Moreover, it traces the exact occupancy situation of the RLC-encoded relaying scenario. Thus, we can update the states in the chain consistently. We now proceed to describe two related schemes useful in getting an insight into the performance of multicast:

- *Oracle-based back-pressure scheme:* Every time two relay nodes establish a link, an “oracle” in the network informs both nodes with the values of $b_{\{i\}^c}$, $b_{\{j\}^c}$, and $b_{\{i,j\}^c}$. The nodes then calculate the mutual occupancies and elect the one that has a higher mutual occupancy to be the transmitter.
- *Blind back-pressure scheme:* When a link is established between any two relay nodes, no information is available to both nodes regarding the mutual occupancies. Hence, one of them is chosen to be the transmitting node at random.

Note that the first scheme is practically impossible to implement, due to the following reason: Take a simple example where node i has four packets p_1, p_2, p_3 , and p_4 and where node j has packets p_5 and p_6 . Moreover, another node that possesses 4 linear combinations of p_1, p_2, p_3 , and p_4 in the mean time, without the knowledge of i or j , delivers four innovative packets to the destination. This means that the innovativeness of node i with respect to node j is actually 0. The second scheme is employed in the multicast protocol described in the previous section. However, some transmissions in this scheme might be useless from the point of view of back-pressure routing. We first show that the first scheme achieves the same performance

as multihop relaying for unicast *without* network coding.

Theorem 10 *The oracle-based scheme for multicast relaying achieves the same per-destination throughput as that of unicast multihop relaying without network coding.*

The proof of the above claim involves looking at the chain from the point of view of a particular relay v and completely enumerating all possible updates to b_S for all S such that $v \in S$. This is possible from the set of operations defined above. Essentially, this process results in the same Markov chain as given in (81)-(85) in Chapter 7. Detailing all the steady-state equations for various cases here would be tedious and counter-intuitive. Hence, we instead describe the intuition behind the proof. We need to observe that both scenarios (i.e., oracle-based multicast and multihop unicast) aim at improving the chance that $b_{[n]} - b_{[n] \setminus \{i\}}$ is non-zero for all $i \in [n]$ whenever a link with established between i and the destination, and the only mechanism that enables this in both cases is relay-to-relay transmissions. The second case (multihop with back pressure) clearly exploits every such opportunity to improve the chance of non-zero occupancy. In the first case (multicast with oracle-based NC), it may first appear that the probability of zero buffer-occupancy on linking with the destination is still lower because of the non-reduction of the occupancy in the node that is transmitting to the other relay. However, this is counter-balanced by the fact that exploiting this lowers the occupancy of the other nodes even when the latter are not involved in a link.

Corollary 8 *The throughput performance of multihop unicast for the given SPMANET model is not improved by employing RLC even with a perfect oracle.*

From the above observations, we conclude that the per-destination performance of multicast communication for our SPMANET model with Random Linear Coding does not perform better than multihop unicast, and that the latter performance can

only be achieved with a perfect oracle that is impossible to implement. We now derive the performance of the practical scheme involving blind relay-to-relay transmission.

8.3.5 Performance of Multicast with Blind Back-Pressure Policy

For the multicast scheme with RLC and blind back-pressure policy, we can yet again describe the entire chain by the $\mathcal{Y}(t)$ -state variables. Here, it is observed that whenever relay-to-relay links occur, the transmission is in the “useful” direction with 50% chance and is in the “useless” direction with 50% chance. We hence observe that its performance will be the same as multihop unicast without RLC wherein the chance that a relay-to-relay link is used is only 50%. Extending our derivation in the previous chapter, we can then derive the performance of the multicast scheme originally described in the chapter by solving the following non-linear steady-state equations iteratively:

Theorem 11 *The steady-state distribution of the states in the collapsed embedded Chain Γ_v is given by the following system of equations. For the sake of convenience, we define $\pi(R_{l_1, l_2})$ to be zero if either l_1 or l_2 lies outside the interval $[0, B]$ in these equations:*

$$\pi(E) = p_{dd} \{\pi(E) + \pi(D_0)\} + p_{rd} \{\pi(R_{0,0}) + \pi(R_{1,0})\} \quad (95)$$

$$\pi(F) = p_{ss} \{\pi(F) + \pi(S_B)\} + p_{rs} \{\pi(R_{B,B}) + \pi(R_{B-1,B})\} \quad (96)$$

$$\pi(D_k) = \begin{cases} p_{dd}\pi(D_{k+1}) + p_{sd}\pi(S_{k+1}) \\ + p_{rd} \left\{ \sum_{j=0}^k \frac{1}{2} \pi(R_{j,k+2}) + \sum_{j'=k}^{k+2} \pi(R_{j',k+1}) \right. \\ \left. + \frac{1}{2} \sum_{j''=k+2}^B \pi(R_{j'',k}) \right\}, 0 \leq k \leq B-2 \\ p_{sd} \{\pi(F) + \pi(S_B)\} + p_{rd} \{\pi(R_{B-1,B}) + \pi(R_{B,B})\}, k = B-1 \end{cases} \quad (97)$$

$$\pi(S_k) = \begin{cases} p_{ds} \{ \pi(E) + \pi(D_0) \} + p_{rs} \{ \pi(R_{1,0}) + \pi(R_{0,0}) \}, k = 1 \\ p_{ss} \pi(S_{k-1}) + p_{ds} \pi(D_{k-1}) \\ + p_{rs} \left\{ \sum_{j=0}^{k-2} \pi(R_{j,k}) + \frac{1}{2} \sum_{j'=k-2}^k \pi(R_{j',k-1}) \right. \\ \left. + \frac{1}{2} \sum_{j''=k}^B \pi(R_{j'',k-2}) \right\}, 1 < k \leq B \end{cases} \quad (98)$$

For any $0 \leq l_1 \leq B$,

$$\varphi_{l_1, l_2}^{-1} \pi(R_{l_1, l_2}) = \begin{cases} p_{dr} \{ \pi(E) + \pi(D_0) \} + p_{rr} \{ \pi(R_{0,0}) + \pi(R_{1,0}) \}, l_2 = 0 \\ p_{sr} \{ \pi(F) + \pi(S_B) \} + p_{rr} \{ \pi(R_{B,B}) + \pi(R_{B-1,B}) \}, l_2 = B \\ p_{sr} \pi(S_{l_2}) + p_{dr} \pi(D_{l_2}) \\ + p_{rr} \left\{ \frac{1}{2} \sum_{j=0}^{l_2-1} \pi(R_{j, l_2+1}) + \sum_{j'=l_2-1}^{l_2+1} \pi(R_{j', l_2}) \right. \\ \left. + \frac{1}{2} \sum_{j''=l_2+1}^B \pi(R_{j'', l_2-1}) \right\}, 0 < l_2 < B \end{cases} \quad (99)$$

where $\varphi_{l_1, l_2} \triangleq \frac{\pi(R_{l_2, l_1})}{\sum_{j=0}^B \pi(R_{j, l_1})}$ for any $0 \leq l_1, l_2 \leq B$.

The above Markov chain has a similar structure after the inclusion of R -states is shown in Fig. 54, where the states that are common with the corresponding two-hop chain are shown in solid bubbles. In this figure, all the possible transitions from various states (or groups of states) is shown. The internal structure of the R -chains vary according to their definitions.

The analysis of steady-state for the Γ_v chain would be complete once we determine the unknown parameter $\alpha(n)$ in terms of networking and mobility parameters. Doing so would enable us to study scaling laws with respect to network size (n), node-density, mobility characteristics, etc. The relevant analysis follows in a manner similar to the previous chapters, and is avoided here due to redundancy.

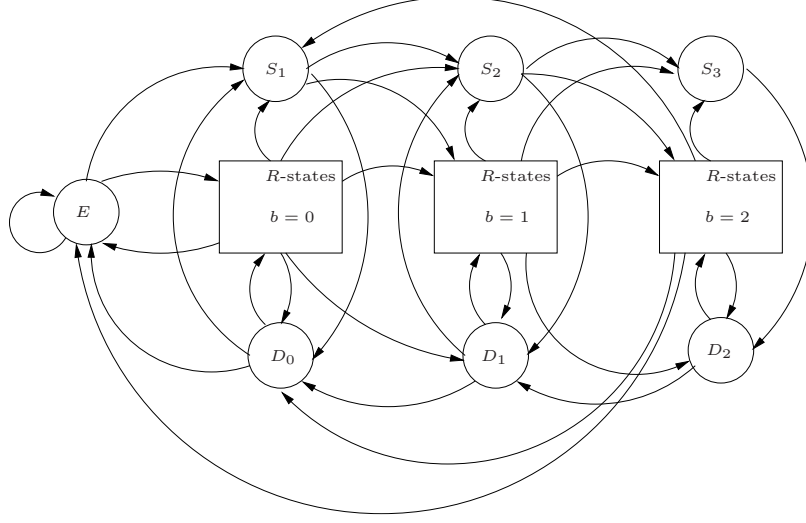


Figure 54: General embedded chain for network-coded multicast.

8.4 Simulation Results

In order to verify our analysis, we simulated the above SPMANET network model under the random-waypoint mobility model. Relay nodes, 50 in number, were deployed on a square region of size $5km$ by $5km$. Each node was assumed to have a radio range consisting of a circular disc of radius $250m$. The velocities of nodes were chosen randomly according to uniform distribution, between $4kmph$ and $9kmph$. Waypoints were randomly chosen from a uniform spatial distribution across the entire deployment region. The throughput performance of such a network at steady-state was obtained for different choices of buffer sizes (B) and for different numbers of destination nodes served (m). We simulated both the network-coded scheme as well as the simple custodial-multicast scheme, and compared the results with the plot obtained by iteratively solving the two Markov Chain described in the previous section. For the network-coded scheme, we used a Galois field of size 397. A prime number was chosen rather than a power of two, since operations are easier in the former case. For both schemes, we simulated 100000 epochs, and repeated the same for 50 different initial conditions.

In the first plot shown in Fig. 55, the per-node buffer size in the network was varied

from 8 to 40 packets, while 10 destinations were served. Under this regime, it is seen that the network-coded scheme offers considerable improvement on the throughput. However, it was observed that the percent improvement in throughput goes down as we increase the buffer sizes further, and they both tend to saturate near the same level. This is due to the fact that the benefits of network coding are contributed primarily by its buffer-management strategy.

In the second plot shown in Fig. 56, the buffer sizes were kept constant at 16 packets per node while m was varied from 5 through 20. Clearly, the higher the number of destinations served, the simple custodial scheme degrades drastically in terms of per-destination-node throughput. However, it was observed that the network-coded scheme hardly diminished in performance.

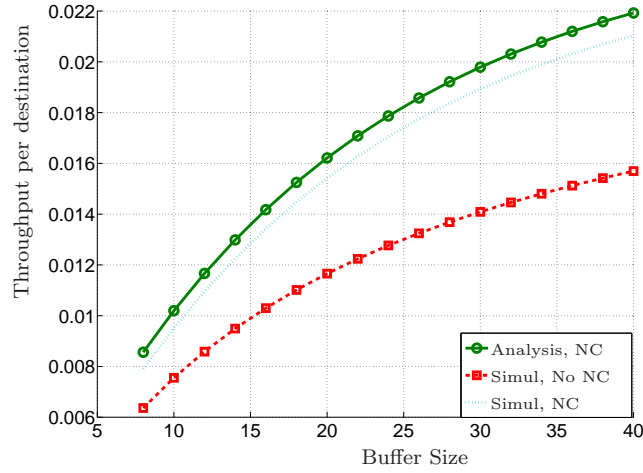


Figure 55: Simulation results for varying buffer sizes

8.5 Summary

We developed a novel methodology to analyze the performance of network-coded multicast in SPMANETs. We presented a generalized, scalable iterative framework based on Markov-chain analysis that incorporates several considerations such as finite buffers, generalized mobility, and node-to-node contention. Our results show

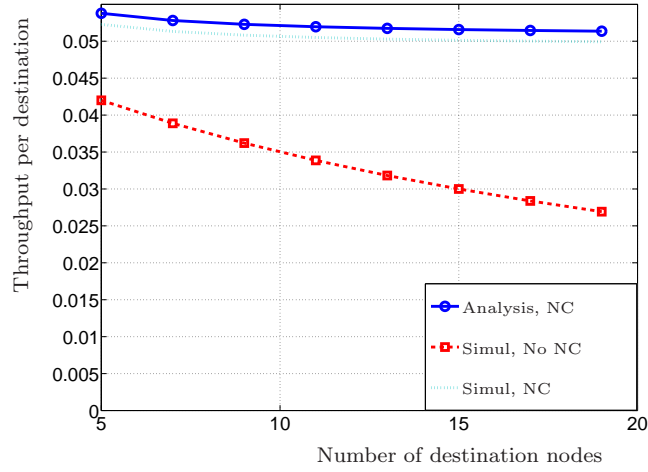


Figure 56: Simulation results for varying number of destinations

that Network Coding offers significant benefits for multicast especially in the finite-buffer regime vis-a-vis simpler custodial-multicast schemes. The Markov-chain-based methodology we developed provided good estimates for performance under network coding.

CHAPTER IX

CONCLUSION OF THE THESIS

In this dissertation, we have investigated several aspects and methodologies involved in the performance modeling and study of collaborative wireless ad-hoc and sensor networks. The dissertation spans a broad range of issues in the design of such networks, including energy-efficiency, latency, capacity, routing, and resource management. In the following sections, we summarize the contributions of the thesis.

The first part of the thesis was aimed at addressing performance estimation and design issues in distributed wireless sensor networks, with particular emphasis on lifetime maximization and latency estimation. The first problem that we considered in this direction was the issue of lifetime maximization in a typical sensor networks with uniform random deployment of nodes. This task was aimed at establishing tight analytical bounds on the sleeping probabilities of nodes and on the achievable lifetime of wireless sensor networks in a generic setting. The lifetime of such a network was shown to be constrained by the maximum sleeping-active duty-cycle ratio needed for maintaining the functionality of the network. Further, an energy efficient deployment scheme was suggested in which the node-power depletion is fairly uniformly distributed throughout the network. This scheme made use of the availability of low power auxiliary channel listening radio. Using a combined deployment and sleep-scheduling approach, we showed that an improvement in lifetime by a factor of $O(\sqrt{\frac{n}{\log n}})$ over uniform distribution of nodes is achievable, where n is the number of nodes in the network. We also showed that the throughput capacity of the network is also improved by the same factor by this scheme. Further, we showed that the maximum lifetime of the network is bounded above by $O\left(\frac{n^{3/2}}{\sqrt{\log n}}\right)$.

The next problem that we considered in the realm of distributed sensor networks was that of obtaining the tradeoff between latency and energy-efficiency in sensor networks characterized by convergecast communication. We considered an application model wherein events occur as a Poisson process in time, with the location of the event being uniform throughout the deployment area. We further assumed that sensor nodes go to sleep and wake up independently as in a Markov process. We obtained analytical bounds for the average latency for reporting an event, as a function of the location of the event with respect to the sink, as well as a function of several network parameters such as network size, radio range, data rate, duty cycle ratio and wireless bandwidth. A detailed analysis was provided for the case when nodes were deployed in a regular grid. We also provided similar analysis for the case when nodes are deployed according to a uniform random process.

The second part of the thesis focused on Delay-Tolerant Networks and general Sparse Mobile Ad-Hoc Networking, wherein nodes use the *store, carry, and forward* paradigm of communication, rather than rely on the establishment of multihop paths for communication as in general MANETs. The problems that we addressed involved those of modeling the throughput capacity of such networks under various communication schemes. The first problem that we considered was that of throughput estimation for simple two-hop unicast with a single pair of immobile source/destination nodes deployed on a grid that communicate with the aid of several mobile relay nodes. These relay nodes were assumed to possess a finite buffer space to store, carry, and forward packets from the source node to the destination node. The mobility model considered here involved a simple random-walk process. To the best of our knowledge, our work was the first to consider finite-buffer effects into throughput-capacity calculations for mobile networks. Even for such an idealistic scheme, we showed that the method Poisson-process approximations for contact durations that is commonly used

in the analysis of intermittently-connected networks with mobility, is highly inaccurate for throughput estimations. Hence, we devise a Markov-chain-based approach that captures the dynamics of the network as a stochastic process. We obtained the throughput capacity of this scheme at steady state as an exact formulation, and evolved a model for incorporating relay-to-relay contention.

On the other hand, we found that this approach, though giving useful insights, is not adaptable to any given scheme. In particular, we observed that the complexity of the Markov chain to be analyzed increases many-fold once we relax the criterion that the source/destination nodes need to be immobile. This is especially because the approach tries to formulate a Markov chain directly from the state-variable of each node which has multiple parameters. We then developed a novel approach based on embedded Markov chains and queuing theory that overcomes the above issues. As the first outcome of this effort, we were able to extend the analysis of simple two-hop unicast from grid networks to networks with general stochastic mobility behaviors. This is a significant contribution of our work, as one is always interested in investigating whether a given communication protocol is robust to changes in mobility patterns. We then proceeded to extend this framework to other communication scenarios. The first case that was taken up was the multiple-unicast case, wherein several unicast streams compete for the same resource in the network. We proposed two different buffer-management strategies for the same, and established by theory that the floating-buffer approach always outperforms the fixed-buffer-segments policy.

The next problem that we solved was that of multihop routing. Though we had considered only two-hop routing up to that point, we intended to investigate whether multi-hop routing improves performance, and whether it can be achieved at a reasonable cost. For both these questions, we found that a judicious back-pressure buffer-management policy improves the throughput performance under limited-buffer scenarios while the cost does not grow undesirably with the gain. We then extended

the above generalized Markov-chain-based approach to this scenario and obtained a low-complexity embedded chain that was solved iteratively in a scalable manner.

The last scenario that we considered for the analysis of SPMANETs was that of network-coded multicast. In particular, we were interested in understanding how this scheme compares with traditional approaches such as custodial multicast. We found that though solving network-coded multicast in an exact manner using the above framework was infeasible, the framework was quite useful in providing two tight (upper and lower) bounds on the performance of this scheme. Further, these bounds are obtained from two Markov chains similar in structure to the multihop case. The outcome of this study was that we found the network-coded scheme to be highly resilient to any increase in the number of destination nodes served, and does not degrade in performance beyond the natural degradation due to increase in contention for the wireless channel between the destination nodes. This is a highly encouraging result, as it could spur further activity in the research community on the application of Network Coding to practical wireless networks.

APPENDIX A

DERIVATION OF THE MEAN AND VARIANCE OF THE HOPWISE EUCLIDEAN ADVANCE (SECTION 3.3)

In this section, we shall compute the probability distribution function of the next-hop advance, and the related parameters, used in Section 3.3. In the Geographical Routing protocol that we employed in Section 3.3, a single node among all the “feasible” ones is randomly chosen to be the next-hop. Firstly, one has to note that the nodes themselves are distributed randomly according to uniform distribution throughout the field. Hence, the joint density function of the coordinate positions X and Y of a feasible next-hop is

$$f_{XY}(X, Y) = \frac{3}{\pi r^2(n)} \quad (100)$$

Here, a little observation reveals that X has the statistics of the next-hop advance parameter used in Section 3.3. Then, the required probability density function of X can be found from the joint pdf, and the support for X , as shown in Fig 57, is the region such that any node within it can be a feasible next hop neighbor.

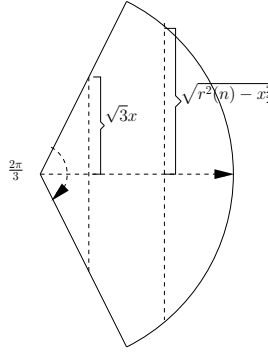


Figure 57: Computation of Next-Hop Advance PDF

$$\begin{aligned}
f_X(x) &= \int_{\text{Support } y|x} f_{XY}(x, y) dy \\
&= \begin{cases} \frac{3}{\pi r^2(n)} \int_{-\sqrt{3}x}^{\sqrt{3}x} dx & \text{if } 0 \leq x \leq \frac{r(n)}{2} \\ \frac{3}{\pi r^2(n)} \int_{-\sqrt{r^2(n)-x^2}}^{\sqrt{r^2(n)-x^2}} dx & \text{if } \frac{r(n)}{2} \leq x \leq r(n) \end{cases} \\
&= \begin{cases} \frac{6}{\pi r^2(n)} \sqrt{3}x & \text{if } 0 \leq x \leq \frac{r(n)}{2} \\ \frac{6}{\pi r^2(n)} \sqrt{r^2(n) - x^2} & \text{if } \frac{r(n)}{2} \leq x \leq r(n) \end{cases} \tag{101}
\end{aligned}$$

In order to apply the Central Limit Theorem on i.i.d. X , we need to find the mean ν and the standard deviation σ .

$$\begin{aligned}
\nu = E(X) &= \int_0^{\frac{r(n)}{2}} \frac{6\sqrt{3}}{\pi r^2(n)} x^2 dx + \int_{\frac{r(n)}{2}}^{r(n)} \frac{6}{\pi r^2(n)} x \sqrt{r^2(n) - x^2} dx \\
&= \frac{\sqrt{3}}{4\pi} r(n) + \frac{6}{\pi r^2(n)} \left\{ \left[\frac{x^2}{2} \sqrt{r^2(n) - x^2} \right]_{\frac{r(n)}{2}}^{r(n)} + \int_{\frac{r(n)}{2}}^{r(n)} \frac{x^2}{2} \frac{x}{\sqrt{r^2(n) - x^2}} dx \right\} \\
&= \frac{\sqrt{3}}{4\pi} r(n) + \frac{6}{\pi r^2(n)} \left\{ -\frac{r^2(n)}{8} \frac{\sqrt{3}r(n)}{2} + \frac{1}{4} \int_{\frac{r^2(n)}{4}}^{r^2(n)} \frac{z}{\sqrt{r^2(n) - z}} dz \right\} \\
&= -\frac{\sqrt{3}}{8\pi} r(n) + \frac{6}{\pi r^2(n)} \frac{1}{4} \int_{r^2(n)/4}^{r^2(n)} \left(-\sqrt{r^2(n) - z} + \frac{r^2(n)}{r^2(n) - z} \right) dz \\
&\quad -\frac{\sqrt{3}}{8\pi} r(n) + \frac{3}{2\pi r^2(n)} \left[-2r^2(n) \sqrt{r^2(n) - z} \right. \\
&= \quad \left. + \frac{2}{3} \left\{ (r^2(n) - z) \sqrt{r^2(n) - z} \right\} \right]_{\frac{r^2(n)}{4}}^{r^2(n)} \\
&= -\frac{\sqrt{3}}{8\pi} r(n) + \frac{9\sqrt{3}}{8\pi} r(n) = \frac{\sqrt{3}}{\pi} r(n) \tag{102}
\end{aligned}$$

$$\begin{aligned}
\nu^2 + \sigma^2 = E(X^2) &= \int_0^{\frac{r(n)}{2}} \frac{6\sqrt{3}x}{\pi r^2(n)} x^2 dx + \int_{\frac{r(n)}{2}}^{r(n)} \frac{6}{\pi r^2(n)} \left(\sqrt{r^2(n) - x^2} \right) x^2 dx \\
&= \frac{6\sqrt{3}}{\pi r^2(n)} \frac{1}{4} \frac{r^4(n)}{16} + \frac{6}{\pi r^2(n)} \int_{\frac{\pi}{6}}^{\frac{\pi}{2}} r^2(n) \sin^2 \theta \times \cos \theta d\theta \\
&= \frac{3\sqrt{3}}{32\pi} r^2(n) + \frac{6}{\pi} r^2(n) \int_{\frac{\pi}{6}}^{\frac{\pi}{2}} \frac{1 - \cos \theta}{8} d\theta \\
&= \frac{3\sqrt{3}}{32\pi} r^2(n) + \frac{3}{4\pi} r^2(n) \left\{ \frac{\pi}{3} - \frac{1}{4} \left(0 - \frac{\sqrt{3}}{2} \right) \right\} \\
&= \frac{3\sqrt{3}}{16\pi} r^2(n) + \frac{1}{4} r^2(n) \tag{103}
\end{aligned}$$

$$\Rightarrow \sigma = \sqrt{\frac{3\sqrt{3}}{16\pi} + \frac{1}{4} - \frac{3}{\pi^2}} r(n) \tag{104}$$

APPENDIX B

PROOF OF LEMMA 5

For simplicity, let us assume that $m' = 1$ in the mobility model. The proof is similar in general, but we take this particular case for clarity. Let $p_{(a_1, a_2)}(\tau)$ be the probability distribution function of the inter-contact times. In other words, $p_{(a_1, a_2)}(\tau)$ is the probability measure of the following event: Given that the pair (a_1, a_2) is in contact, the next contact for this pair occurs exactly after τ . Clearly, $\sum_{\tau=1}^{\infty} p_{(a_1, a_2)}(\tau) = 1$. Now, let \mathbf{p} be the probability distribution of the mobility state of the particular node a_3 when a particular instance of contact between (a_1, a_2) occurs. Clearly, the distribution of node a_3 's position at the very next instance of contact between (a_1, a_2) is given by $\sum_{\tau=1}^{\infty} p_{(a_1, a_2)}(\tau) \Psi_{mob}^{\tau}[\mathbf{p}]$. In other words, if we view samples of the distribution of location of node c taken whenever an (a_1, a_2) -contact occurs, then the state transitions of successive samples are described by the last expression. The proof is complete from the fact that the steady-state distribution of the sampled process is the same, since

$$\sum_{\tau=1}^{\infty} p_{(a_1, a_2)}(\tau) \Psi_{mob}^{\tau}[\boldsymbol{\pi}_{mob}] = \sum_{\tau=1}^{\infty} p_{(a_1, a_2)}(\tau) \boldsymbol{\pi}_{mob} = \boldsymbol{\pi}_{mob}$$

Clearly, $\Psi_{mob}^{\tau}[\boldsymbol{\pi}_{mob}] = \boldsymbol{\pi}_{mob}$, since $\boldsymbol{\pi}_{mob}$ is the steady-state probability distribution for the state transformation functional $\Psi_{mob}(\cdot)$.

APPENDIX C

PROOF OF LEMMA 11

Note that (69) is already proved in the discussion immediately preceding the lemma, as in (68). The best way to show (70)-(73) is by looking at the states of the $\Gamma_{\tau, v'}$ -chain, writing down the steady-state equations, and verifying by back-substitution. We will illustrate this process for the states $\mathcal{D}_{0, b_1, \dots, b_{m-l}, 0, \dots, 0}^{(1)}$. Let us first write $p = \pi(\mathcal{D}_{0, 0, \dots, 0}^{(1)})$, and $p' = \pi(\mathcal{E}_{0, 0, \dots, 0}^{(1)})$. First, note that from (68) we have:

$$p + p' = \frac{m\alpha}{m(1-\alpha) + (m-1)}p' + p' = \frac{2m-1}{m(1-\alpha) + m-1}p' = 2p\gamma^{-1}. \quad (105)$$

The case of $\mathcal{D}_{0, b_1, \dots, b_{m-l}, 0, \dots, 0}^{(1)}$ needs to be bifurcated into two (it will be clear from Case (i) why this is required):

Case (i): $\sum b_i < B - 1$

Consider the state $\mathcal{D}_{0, b_1, \dots, b_{m-l}, 0, \dots, 0}^{(1)}$. In this state, the relay v' linked with the destination number 1, initially had one block of packets in its buffer corresponding to the source/destination pair 1, but delivered it to the destination and erased the same after the epoch. The previous link-situation (i.e., the previous state in $\Gamma_{\tau, v'}$) could have been any of the following:

- a. Link with the same node, i.e., destination #1, resulting in exactly one block corresponding to pair #1 remaining in its buffer. This corresponds to the state $\mathcal{D}_{1, b_1, \dots, b_{m-l}, 0, \dots, 0}^{(1)}$ (Note that this state exists only if $\sum b_i$ is strictly less than $B - 1$).
- b. Link with source #1, resulting in 1 block corresponding to pair #1. This corresponds to the state $\mathcal{S}_{1, b_1, \dots, b_{m-l}, 0, \dots, 0}^{(1)}$.

- c. Link with any destination node $j_1 \in \{m-l+2, m-l+3, \dots, m\}$, one block sent to that destination, resulting in buffer occupancy $(1, b_1, \dots, b_{m-l}, 0, \dots, 0)$. This corresponds to the states $\mathcal{D}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(j_1)}$, $l-1$ in number.
- d. Same as before, but no packets could be sent since there were none corresponding to that particular destination, to start with. This corresponds to the states $\mathcal{E}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(j_2)}$, with $j_2 \in \{m-l+2, m-l+3, \dots, m\}$, $l-1$ in number.
- e. Link with the source/destination node of any pair $j_3 \in \{2, \dots, m-l+1\}$, resulting in buffer occupancy $(1, b_1, \dots, b_{m-l}, 0, \dots, 0)$ where one block was sent/received to the source/destination node number j_3 . These correspond to the classes of states $\mathcal{S}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(j_3)}$ and $\mathcal{D}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(j_3)}$, which are $m-l$ in number each. Note that the latter class exists only if $\sum b_i$ is strictly less than $B-1$.

From the definition of α , note that the probability of reaching state $\mathcal{D}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(1)}$ from case (a.) above is exactly $1-\alpha$. Furthermore, a corollary from the proof of Lemma 10 is that the mobility state distributions of each of the other nodes are identical during a particular contact. This means that the probabilities of reaching $\mathcal{D}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(1)}$ starting from the states in cases (b.)-(e.) are all equal to $\frac{\alpha(m)}{2m-1}$.

Moving on, we know from Lemma 9 that the steady-state probabilities of individual states in case (c.) are all equal to $\pi\left(\mathcal{D}_{0,1,b_1,\dots,b_{m-l},1,0,\dots,0}^{(1)}\right)$; those in case (d.) are all equal to $\pi\left(\mathcal{E}_{0,1,b_1,\dots,b_{m-l},1,0,\dots,0}^{(1)}\right)$; those in case (e.) are all equal to $\pi\left(\mathcal{D}_{b_{j_3},1,b_1,\dots,b_{j_3-1},b_{j_3+1},\dots,b_{m-l},0,\dots,0}^{(1)}\right)$, using (71). Thus, we can write the steady-state

equation for $\mathcal{D}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(1)}$ as:

$$\begin{aligned}
\pi \left(\mathcal{D}_{0,b_1,\dots,b_{m-l},0,\dots,0}^{(1)} \right) &= (1 - \alpha) \pi \left(\mathcal{D}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(1)} \right) + \frac{\alpha}{2m-1} \pi \left(\mathcal{S}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(1)} \right) \\
&\quad + \frac{\alpha}{2m-1} (l-1) \left\{ \pi \left(\mathcal{D}_{0,1,b_1,\dots,b_{m-l},1,0,\dots,0}^{(1)} \right) \right. \\
&\quad \quad \left. + \pi \left(\mathcal{E}_{0,1,b_1,\dots,b_{m-l},1,0,\dots,0}^{(1)} \right) \right\} \\
&\quad + \frac{\alpha}{2m-1} (m-l) \left\{ \pi \left(\mathcal{S}_{b_{j_3},1,b_1,\dots,b_{j_3-1},b_{j_3+1},\dots,b_{m-l},0,\dots,0}^{(1)} \right) \right. \\
&\quad \quad \left. + \pi \left(\mathcal{D}_{b_{j_3},1,b_1,\dots,b_{j_3-1},b_{j_3+1},\dots,b_{m-l},0,\dots,0}^{(1)} \right) \right\}
\end{aligned} \tag{106}$$

Finally, from (70)-(73), we observe that the steady-state probabilities corresponding to cases (a.), (b.), and (e.) are equal to $\gamma^{m-l}p$ each, that for case (c.) are equal to $\gamma^{m-l+1}p$ each, and that for case (d.) are equal to $\gamma^{m-l+1}p'$ each. Substituting these values in (106), we will end up with the already-proved equation in (105). *Case (ii):* $\sum b_i = B - 1$ This is similar to Case (i). The steady-state equation is provided below as a guide to complete the proof:

$$\begin{aligned}
\frac{2m-1}{\alpha} \pi \left(\mathcal{D}_{0,b_1,\dots,b_{m-l},0,\dots,0}^{(1)} \right) &= \left\{ \pi \left(\mathcal{S}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(1)} \right) + \pi \left(\mathcal{F}_{1,b_1,\dots,b_{m-l},0,\dots,0}^{(1)} \right) \right\} \\
&\quad + \left\{ \pi \left(\mathcal{E}_{0,\dots,0,1,b_1,\dots,b_{m-l}}^{(1)} \right) + \pi \left(\mathcal{F}_{0,\dots,0,1,b_1,\dots,b_{m-l}}^{(1)} \right) \right\} (l-1) \\
&\quad + \left\{ \pi \left(\mathcal{S}_{b_1,1,b_2,\dots,b_{m-l},0,\dots,0}^{(1)} \right) + \pi \left(\mathcal{F}_{b_1,1,b_2,\dots,b_{m-l},0,\dots,0}^{(1)} \right) \right\} (m-l)
\end{aligned} \tag{107}$$

REFERENCES

- [1] Piyush Gupta and P. R. Kumar, “The capacity of wireless networks,” *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, 2000.
- [2] Ashish Agarwal and P. R. Kumar, “Improved capacity bounds for wireless networks,” *Wireless Communications and Mobile Computing*, vol. 4, no. 3, pp. 251–261, 2004.
- [3] H. Pishro-Nik, K. Chan, , and F. Fekri, “On connectivity properties of large-scale wireless sensor networks,” *Proc. First Annual Communications Society Conf. on Sensor Communications and Networks Santa Clara*, pp. 498–507, 4-7 October 2004.
- [4] Sanjay Shakkottai, Rayadurgan Srikant, and Ness B. Shroff, “Unreliable sensor grids: coverage, connectivity and diameter,” *Ad Hoc Networks*, vol. 3, no. 6, pp. 702–716, 2005.
- [5] Abbas El Gamal, James P. Mammen, Balaji Prabhakar, and Devavrat Shah, “Optimal throughput-delay scaling in wireless networks: part I: the fluid model,” *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2568–2592, 2006.
- [6] Matthias Grossglauser and David N. C. Tse, “Mobility increases the capacity of ad hoc wireless networks,” *IEEE/ACM Trans. Netw.*, vol. 10, no. 4, pp. 477–486, 2002.
- [7] Abbas El Gamal, James P. Mammen, Balaji Prabhakar, and Devavrat Shah, “Throughput-delay trade-off in wireless networks,” *INFOCOM*, 2004.
- [8] K.S. Chan, H. Pishro-Nik, and F. Fekri, “Analysis of hierarchical algorithms for wireless sensor network routing protocols,” *Wireless Communications and Networking Conference, 2005 IEEE*, vol. 3, pp. 1830–1835 Vol. 3, March 2005.
- [9] Ian F. Akyildiz, W. Su, Yogesh Sankarasubramaniam, and Erdal Cayirci, “Wireless sensor networks: a survey,” *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [10] Curt Schurgers, Vlasios Tsiatsis, Saurabh Ganeriwal, and Mani B. Srivastava, “Topology management for sensor networks: exploiting latency and density,” *MobiHoc*, pp. 135–145, 2002.
- [11] Zhao Qun and Mohan Gurusamy, “Topology knowledge range control for lifetime maximization in sensor networks with data aggregation,” *PE-WASUN*, pp. 84–91, 2005.

- [12] Tommaso Melodia, Dario Pompili, and Ian F. Akyildiz, "Optimal local topology knowledge for energy efficient geographical routing in sensor networks.," *INFOCOM*, 2004.
- [13] Wei Ye, John S. Heidemann, and Deborah Estrin, "An energy-efficient mac protocol for wireless sensor networks.," *INFOCOM*, 2002.
- [14] Tijs van Dam and Koen Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks.," *SenSys*, pp. 171–180, 2003.
- [15] Wei Ye, John S. Heidemann, and Deborah Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks.," *IEEE/ACM Trans. Netw.*, vol. 12, no. 3, pp. 493–506, 2004.
- [16] Stephanie Lindsey, Cauligi S. Raghavendra, and Krishna M. Sivalingam, "Data gathering in sensor networks using the *energy delay* metric.," *IPDPS*, p. 188, 2001.
- [17] Stephanie Lindsey, Cauligi S. Raghavendra, and Krishna M. Sivalingam, "Data gathering algorithms in sensor networks using energy metrics.," *IEEE Trans. Parallel Distrib. Syst.*, vol. 13, no. 9, pp. 924–935, 2002.
- [18] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks.," *HICSS*, 2000.
- [19] Wei Ye, John Heidemann, and Deborah Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *INFOCOM 2002: Proceedings of the Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*. IEEE, June 2002, vol. 3, pp. 1567–1576.
- [20] Tijs van Dam and Koen Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, New York, NY, USA, 2003, pp. 171–180, ACM Press.
- [21] Rong Zheng and Robin Kravets, "On-demand power management for ad hoc networks," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*, San Francisco, March 2003, IEEE, vol. 1.
- [22] Wei Ye, John Heidemann, and Deborah Estrin, "Medium access control with coordinated, adaptive sleeping for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 3, pp. 493–506, June 2004.
- [23] Yang Yu, Bhaskar Krishnamachari, and Viktor K. Prasanna, "Energy-latency tradeoffs for data gathering in wireless sensor networks.," *INFOCOM*, 2004.

- [24] M. Borghini, F. Cuomo, T. Melodia, U. Monaco, and F. Ricciato, “Optimal data delivery in wireless sensor networks in the energy and latency domains,” *wicon*, vol. 00, pp. 138–145, 2005.
- [25] Curt Schurgers, Vlasios Tsiatsis, Saurabh Ganeriwal, and Mani B. Srivastava, “Optimizing sensor networks in the energy-latency-density design space,” *IEEE Trans. Mob. Comput.*, vol. 1, no. 1, pp. 70–80, 2002.
- [26] Tai-Lin Chin, Parameswaran Ramanathan, and Kewal K. Saluja, “Analytic modeling of detection latency in mobile sensor networks,” *IPSN*, pp. 194–201, 2006.
- [27] Olivier Dousse, Petteri Mannersalo, and Patrick Thiran, “Latency of wireless sensor networks with uncoordinated power saving mechanisms,” *MobiHoc*, pp. 109–120, 2004.
- [28] Santosh Kumar, Ten-Hwang Lai, and József Balogh, “On k-coverage in a mostly sleeping sensor network,” *MOBICOM*, pp. 144–158, 2004.
- [29] David B Johnson and David A Maltz, “Dynamic source routing in ad hoc wireless networks,” *Mobile Computing*, vol. 353, 1996.
- [30] S. Merugu, M. Ammar, and E. Zegura, “Routing in space and time in networks with predictable mobility,” *Technical Report GIT-CC-04-7, Georgia Institute of Technology.*, 2004.
- [31] Wenrui Zhao and Mostafa H. Ammar, “Message ferrying: Proactive routing in highly-partitioned wireless ad hoc networks,” *9th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS 2003)*, pp. 308–314, 2003.
- [32] Ling-Jyh Chen, Chen-Hung Yu, Tony Sun, Yung-Chih Chen, and Hao hua Chu, “A hybrid routing approach for opportunistic networks,” *CHANTS ’06: Proceedings of the 2006 SIGCOMM workshop on Challenged networks*, pp. 213–220, 2006.
- [33] Philo Juang, Hidekazu Oki, Yong Wang, Margaret Martonosi, Li Shiuan Peh, and Daniel Rubenstein, “Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet,” *SIGOPS Oper. Syst. Rev.*, vol. 36, no. 5, pp. 96–107, 2002.
- [34] Ramanan Subramanian and Faramarz Fekri, “Sleep scheduling and lifetime maximization in sensor networks: fundamental limits and optimal solutions,” *IPSN*, pp. 218–225, 2006.
- [35] Ramanan Subramanian and Faramarz Fekri, “Bounds for lifetime optimization with guaranteed information delivery in convergecast sensor networks,” *Elsevier Ad-Hoc Networks*, under revision, revised July 2009.

- [36] M. Penrose, *Random Geometric Graphs*, Oxford University Press, 2003.
- [37] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inform. Theory*, vol. 46, no. 2, pp. 388–404, 2000.
- [38] L. C. Zhong C. Guo and J. M. Rabaey, "Low power distributed mac for ad hoc sensor radio networks," in *Proceedings of IEEE GlobeCom 2001*, San Antonio, November 2001, IEEE, vol. 5, pp. 2944–2948.
- [39] Matthew J. Miller and Nitin H. Vaidya, "Minimizing energy consumption in sensor networks using a wakeup radio," in *Wireless Communications and Networking Conference (WCNC 2004)*. IEEE, March 2004, vol. 4, pp. 2335–2340.
- [40] Brad Karp and H. T. Kung, "Gpsr: greedy perimeter stateless routing for wireless networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*, New York, NY, USA, 2000, pp. 243–254, ACM Press.
- [41] R. Subramanian and F. Fekri, "Analysis of latency and related tradeoffs in distribute sensor networks," *Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON '07. 4th Annual IEEE Communications Society Conference on*, pp. 391–400, June 2007.
- [42] R. Subramanian and F. Fekri, "Modeling and analysis of latency in distributed sensor networks for converge-cast traffic," *IEEE Transactions on Wireless Networks*, under revision, revised March 2009.
- [43] Michael S Jacobson, André E. Kézdy, Ewa Kubicka, Kubicki Grzegorz, Jenő Lehel, Chi Wang, and Douglas B West, "The path spectrum of a graph," *Proceedings of the Twenty-sixth Southeastern International Conference on Combinatorics, Graph Theory and Computing*, vol. 112, pp. 49–63, 1995.
- [44] Kevin R. Fall, "A delay-tolerant network architecture for challenged internets," *Proceedings of the ACM SIGCOMM 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 27–34, August 25-29 2003.
- [45] Charles Perkins and Pravin Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, pp. 234–244, 1994.
- [46] Sushant Jain, Kevin R. Fall, and Rabin Patra, "Routing in a delay tolerant network.," *Proceedings of the ACM SIGCOMM 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 145–158, 2004.

- [47] Yong Wang, Sushant Jain, Margaret Martonosi, and Kevin Fall, "Erasure-coding based routing for opportunistic networks," *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, pp. 229–236, 2005.
- [48] J. Widmer and J. Y. Le Boudec, "Network coding for efficient communication in extreme networks," *Proc. of the ACM SIGCOMM workshop on Delay-tolerant networking*, 2005.
- [49] Hyewon Jun, Mostafa Ammar, and Ellen Zegura, "Power management in delay tolerant networks: A framework and knowledge-based mechanisms," *2nd IEEE Conference on Sensor and Ad Hoc Communication and Networks*, Sept 2005.
- [50] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2004)*, pp. 187–198, May 2004.
- [51] Muhammad Mukarram Bin Tariq, Mostafa H. Ammar, and Ellen W. Zegura, "Message ferry route design for sparse ad hoc networks withm obile nodes," *MobiHoc*, pp. 37–48, 2006.
- [52] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," 2000.
- [53] A. Lindgren, A. Doria, and O. Scheln, "Probabilistic routing in intermittently connected networks," 2003.
- [54] Sushant Jain, Michael J. Demmer, Rabin Patra, and Kevin R. Fall, "Using redundancy to cope with failures in a delay tolerant network.," *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pp. 109–120, 2005.
- [55] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi Raghavendra, "Efficient routing in intermittently connected mobile networks: The single-copy case," *ACM/IEEE Transactions on Networking*, 2007, to appear.
- [56] Apoorva Jindal and Konstantinos Psounis, "Contention-aware analysis of routing schemes for mobile opportunistic networks," *MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*, pp. 1–8, 2007.
- [57] Robin Groenevelt, Philippe Nain, and Ger Koole, "The message delay in mobile ad hoc networks," *Perform. Eval.*, vol. 62, no. 1-4, pp. 210–228, 2005.
- [58] A. Al-Hanbali, A. A. Kherani, and P. Nain, "Simple models for performance evaluation of a class of two-hop relay protocols," .

- [59] Ahmad Al Hanbali, Philippe Nain, and Eitan Altman, "Performance of ad hoc networks with two-hop relay routing and limited packet lifetime," *valuetools '06: Proceedings of the 1st international conference on Performance evaluation methodologies and tools*, p. 49, 2006.
- [60] Ramanan Subramanian and Faramarz Fekri, "Unicast throughput of finite-buffer sparse mobile networks," *Proceedings of the Forty-Sixth Annual Allerton Conference*, September 23–26 2008, to appear.
- [61] John Burgess, Brian Gallagher, David Jensen, and Brian Neil Levine, "Max-prop: Routing for vehicle-based disruption-tolerant networks," *In Proc. IEEE INFOCOM*, 2006.
- [62] Matthew Seligman, Kevin Fall, and Padma Mundur, "Storage routing for DTN congestion control," *Wireless Communications and Mobile Computing*, vol. 7, no. 10, pp. 1183–1196, 2007.
- [63] Luigi Liquori, Diego Borsetti, Claudio Casetti, and Carla-Fabiana Chiasserini, "An overlay architecture for vehicular networks," *Networking*, pp. 60–71, 2008.
- [64] Hongyi Wu, Yu Wang, Ha Dang, and Feng Lin, "Analytic, simulation, and empirical evaluation of delay/fault-tolerant mobile sensor networks," *IEEE Transactions on Wireless Communications*, vol. 6, no. 9, pp. 3287–3296, 2007.
- [65] "Kapsch on board unit obu3021 specifications," <http://www.kapsch.com/TrafficCom/en/files/KTC-OBU3021.pdf>.
- [66] Xiaolan Zhang, Giovanni Neglia, Jim Kurose, and Don Towsley, "Performance modeling of epidemic routing," *Comput. Netw.*, vol. 51, no. 10, pp. 2867–2891, 2007.
- [67] Robert B. Cooper, *Introduction to Queueing Theory (2nd edition)*, Elsevier, 1981.
- [68] Kenneth A. Berman and Mokhtar H. Konsowa, "Random paths and cuts, electrical networks, and reversible markov chains," *SIAM J. Discret. Math.*, vol. 3, no. 3, pp. 311–319, 1990.
- [69] L. Lavatelli, "The Resistive Net and Finite-Difference Equations," *American Journal of Physics*, vol. 40, pp. 1246–1257, Sept. 1972.
- [70] B. Melamed and D. Yao, *Advances in Queueing: Theory, Methods and Open Problems*, chapter The ASTA Property in Queueing, invited chapter, pp. 195–224, CRC Press, 1995.
- [71] R. Subramanian and F. Fekri, "Throughput analysis of delay tolerant networks with finite buffers," *Mobile Ad Hoc and Sensor Systems, 2008. MASS 2008. 5th IEEE International Conference on*, pp. 790–795, 29 2008-Oct. 2 2008.

- [72] Ramanan Subramanian, Badri N. Vellambi, and Faramarz Fekri, "A generalized framework for throughput analysis in sparse mobile networks," *Proc. of the 7th Intl. Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt09)*, 2009.
- [73] R. Subramanian and F. Fekri, "Analysis of throughput capacity in sparse mobile networks: A queueing-theoretic framework," *IEEE Transactions on Information Theory*, submitted April 2009.
- [74] Member-Jungkeun Yoon, Member-Mingyan Liu, and Member-Brian Noble, "A general framework to construct stationary mobility models for the simulation of mobile networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 7, pp. 860–871, 2006.
- [75] Badri N. Vellambi, Ramanan Subramanian, Faramarz Fekri, and Mostafa Ammar, "Reliable and efficient message delivery in delay tolerant networks using rateless codes," *MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*, pp. 91–98, 2007.
- [76] D. Aldous and J. Fill, *Reversible Markov Chains and Random Walks on Graphs*, Monograph in preparation, available at <http://www.stat.berkeley.edu/users/aldous/RWG/book.html>, 2002.
- [77] E. Hyttiä and J. Virtamo, "Random waypoint model in cellular networks," *Wireless Networks*, , no. 2, pp. 177–188, Apr. 2007.
- [78] William Navidi and Tracy Camp, "Stationary distributions for the random waypoint mobility model," *IEEE Transactions on Mobile Computing*, vol. 3, no. 1, pp. 99–108, 2004.
- [79] Esa Hytti, Pasi Lassila, and Jorma Virtamo, "Spatial node distribution of the random waypoint mobility model with applications," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 680–694, 2006.
- [80] Muhammad Mukarram Bin Tariq, Mostafa H. Ammar, and Ellen W. Zegura, "Message ferry route design for sparse ad hoc networks with mobile nodes," *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing, (MobiHoc 2006)*, pp. 37–48, 2006.
- [81] Ramanan Subramanian and Faramarz Fekri, "Analysis of multiple-unicast throughput in finite-buffer delay-tolerant networks," *Information Theory, 2009. ISIT 2009. IEEE International Symposium on*, pp. 1634–1638, 28 2009–July 3 2009.
- [82] R. Subramanian and F. Fekri, "Analysis of multihop routing in a sparse mobile ad-hoc network with finite buffers," *IEEE Transactions on Mobile Computing*, submitted June 2009.

- [83] Yunfeng Lin, Ben Liang, and Baochun Li, “Performance modeling of network coding in epidemic routing,” *MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*, pp. 67–74, 2007.
- [84] Yunfeng Lin, Baochun Li, and Ben Liang, “Stochastic analysis of network coding in epidemic routing,” *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 5, pp. 794–808, June 2008.
- [85] Shirish Karande, Zheng Wang, Hamid Sadjadpour, and J J Garcia-Luna-Aceves, “Network coding does not change the multicast throughput order of wireless ad hoc networks,” *Proceedings of ICC'09*, June 2009.
- [86] Shirish Karande, Zheng Wang, Hamid Sadjadpour, and J J Garcia-Luna-Aceves, “Multicast throughput order of network coding in wireless ad-hoc networks,” *Proceedings of SECON'09*, June 2009.
- [87] Susan Symington, Robert C. Durst, and Keith Scott, “Custodial multicast in delay tolerant networks,” *Consumer Communications and Networking Conference, 2007. CCNC 2007. 4th IEEE*, pp. 207–211, Jan. 2007.

VITA

Ramanan Subramanian received his Bachelor of Technology (B.Tech) degree in Electrical Engineering from the Indian Institute of Technology (Madras), India in 2003. Since the Spring of 2004, he has been enrolled in the graduate degree program of the School of Electrical and Computer Engineering at the Georgia Institute of Technology, Atlanta, USA. Since then, he earned his Master of Science in July 2006, and obtained his Doctor of Philosophy in December 2009, both from Georgia Tech. He will be joining as a Research Fellow in the Institute of Telecommunications Research affiliated with the University of South Australia in Adelaide, AU. Ramanan's research interests include telecommunications, wireless networking, abstract modeling of networks, queuing theory, and statistics. He was awarded the "Outstanding Research Award for the Spring of 2009" by the Center for Signal and Image Processing (CSIP) at Georgia Tech.