ABSTRACT

Title of Dissertation: DISTRIBUTION OF PATH DURATION IN WIRELESS AD-HOC NETWORKS AND PATH SELECTION

Yijie Han, Doctor of Philosophy, 2006

Dissertation directed by: Professor Richard J. La
Department of Electrical and Computer Engineering

The performance of routing protocols in wireless ad-hoc networks is determined by a number of factors, among which the path durations are of much importance. Path durations affect the reliability of the network service provided to the applications and the routing overhead incurred. In this dissertation, we study the distribution of path duration in wireless ad-hoc networks and its impact on routing. We focus on identifying a scheme that selects a path with the largest expected duration for data transmission when multiple paths are available between a source-destination pair. To this end, we first study the distribution of path duration. Our analytical result also reveals the relation between link level and path level statistics, which can be used to estimate expected path durations.

Our main results show that in a large scale wireless ad-hoc network, as long as the local dependency among link excess lives is not too strong, the path durations

can be well approximated by an exponential random variable for paths with sufficiently large hop count. Furthermore, the inverse of the expected duration of a path can be estimated using the sum of the inverses of the expected durations of the links along the path.

Based on these analytical results, we propose a new path selection scheme referred to as "Maximum Expected Duration" (MED) path selection. This scheme can be easily incorporated into existing routing protocols. Information needed for estimating the expected path duration can be collected during path discovery phase. Our simulation results demonstrate that under routing protocols with the MED path selection scheme added, the median value of the path durations can be increased up to 60 percent compared to those without using the scheme. Moreover, we reduce the delay and overhead during local path recovery by using cached paths that are likely to be available when the primary path breaks down.

DISTRIBUTION OF PATH DURATION IN WIRELESS AD-HOC NETWORKS AND PATH SELECTION

by

Yijie Han

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2006

Advisory Committee:

Professor Richard J. La, Chairman
Professor Armand M. Makowski
Professor Mark Shayman
Professor Nuno Martins
Professor A. Udaya Shankar

DEDICATION

To My Parents

ACKNOWLEDGEMENTS

I owe my gratitude to all the people who have made this thesis possible and because of whom my graduate experience has been one that I will cherish forever.

First and foremost I would like to thank my advisor, Professor Richard La for giving me an invaluable opportunity to work on challenging and extremely interesting projects over the past three years. He has always made himself available for help and advice. I learned tremendously from his insights, working ethics and endless endeavor to make sense of everything.

I would also like to express my deep and sincere gratitude to Professor Armand Makowski for his generous help, continuous encouragement, and incredible guidance.

Thanks are due to Professor Mark Shayman, Professor Nuno Martins and Professor A. Udaya Shankar for agreeing to serve on my thesis committee and for sparing their invaluable time reviewing the manuscript.

I owe my deepest thanks to my family - my mother and father who have always stood by me and guided me through my life, my husband who has shared the feeling along the bumpy road of PhD study. Words cannot express the gratitude I owe them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Mobile Ad-hoc Networks (MANETs) have been the focus of active research in the networking area in recent years. A wireless ad-hoc network can be deployed with no infrastructure and mobile nodes can establish and maintain a network in an autonomous manner. It provides great flexibility for participants of the network to communicate with one another. However, there are many challenges in designing and operating such a network.

We list three of them here: First, due to the frequent change of the participants in ad-hoc network and the topology of ad-hoc networks, no centralized control can be implemented to maintain network functionalities. Nodes have to establish and maintain a functional network themselves. In other words, all nodes or entities participating in the network not only benefit from the network services, but also contribute to the service to other nodes. This can lead to inefficient use of network resources and force the network to operate in a suboptimal manner. Meanwhile, the communication between nodes is often carried out over wireless media, and thus is unreliable due to varying channel conditions. Moreover, when nodes are mobile, on-going traffic can be interrupted if one of the nodes along the path moves out of the transmission range of its upstream or downstream neighbor. In short, there are many differences between MANETs and traditional wireline networks. Thus, the design of its architectures and protocols requires further studies.

1.1 Challenges for routing in MANETs

In a MANET, routing is needed to provide a multi-hop connection between nodes that cannot communicate directly. However, the development of efficient routing protocols is a nontrivial and challenging task because of special characteristics of the MANETs.

First, due to nodes' mobility, a network topology varies with time as the connectivity between nodes changes dynamically. Links between nodes are expected to be set up and torn down much more frequently than in a wireline network. When a path in use fails because of a network topology change, the underlying routing algorithm should enable the switching to an alternative path quickly.

Second, most mobile devices are battery powered. Exchanging unnecessary control messages wastes the nodes' energy. Hence, reducing routing overhead during route discovery and recovery plays an important role in reducing energy consumption and extending the network life time.

Third, the available bandwidth for communication between two nodes is limited and varies due to power constraint, channel fading, noise or interference. If possible, nodes should cooperate and reduce interference to each other's transmission. Recently, researchers proposed interference-aware routing *e.g.,* in [24, 41, 42].

Many routing algorithms have been proposed for mobile ad-hoc networks to deal with these challenges unique to MANETs [21, 33, 35, 36]. There are also some monographs on the survey of routing algorithms in MANETs [34, 43]. In this dissertation, we focus on the study to further improve the performance of these routing protocols by using proper path selection scheme.

1.2 Path selection in MANETs

As already mentioned, frequent path failures cause interruption to the network services and degrade applications' performance, especially of those time-critical ones. In MANETs, a connection between two mobile nodes is more likely to be disrupted than a connection between two fixed nodes. This can be simply caused by nodes moving out of the transmission ranges of each other. A path is broken when one or more of the links along the path become unavailable. When this happens, it takes a non-negligible amount of time to locate the failed link and perform the path recovery. During a path failure, the on-going traffic is interrupted, packets may need to be retransmitted and routing control messages have to be exchanged. In short, it causes much waste in energy and other resources. If the routing algorithm is aware of the mobility of the nodes, and can choose paths with larger expected durations, then it is possible to support the applications with more reliable network service and to reduce the control overhead. From this perspective, we propose to take the *expected* durations of the paths into consideration as well as other requirements on the quality of the paths when performing path selection.

Most existing routing protocols select a path based on some heuristic argument. Dynamic Source Routing (DSR) protocol uses the path hop count as the selection criterion, and chooses a path with the minimum hop count. Ad-hoc On-demand Distance Vector (AODV) routing protocol chooses the path that is discovered first. Associativity Based Routing (ABR) protocol uses maximum average age of the links of a path as the metric when comparing two paths. However, it is not clear how these criteria are related to the *expected* duration of a path. We will review these path selection schemes again in Section 2.4. In this dissertation, we propose a new path selection scheme, which is called a *Maximum Expected Duration* (MED) path selection.

1.3 Distribution of path duration

In order to find the criteria for selecting paths based on their expected durations, we should first understand the statistical properties of link and path durations and their relation. This problem initially appears to be very difficult to approach. This is because the exact distribution of link durations in a MANET depends on the mobility patterns of the nodes, such as their speed ranges, their direction changing rules, their idling periods, etc. Many models have been proposed to model different mobility patterns. More details can be found in a survey paper [7].

Although a study on the statistics of link and path durations in MANETs is difficult, it is strongly motivated. For example, the study of path duration distribution will allow us to approximate the frequency of disruption in network service and resulting overhead. This may further help to evaluate the performance of on-demand routing protocols and the adverse effects of potentially frequent disruptions in network service on the performance of upper layers without running time-consuming detailed simulations.

A simulation study of the distribution of multi-hop path durations under various mobility models has been carried out by Sadagopan *et al.* [40]. Their simulation results showed an interesting observation that the distribution of path duration can be accurately approximated by an *exponential* distribution for paths with a hop count greater than 3 or 4. More surprisingly, this was true for *all* mobility models considered, including Random Waypoint mobility model, Reference Point Group mobility model, Freeway model and Manhattan model. However, the reason for the emergence of an exponential distribution was not well investigated. In this dissertation, we develop a theoretical framework to study the distributions of path and link durations.

1.4 Summary of results

1.4.1 Distributional convergence of path duration

Our main result can be summarized as the following. Under a set of mild assumptions on the distribution of link durations and their dependency structure, the distribution of path duration, when appropriately scaled, converges to an exponential distribution as the number of path hop count increases. The proof of this convergence result also reveals that there is a simple relation between the expected duration of a path and the expected durations of the links along the path. As an implication of this result, for a path with a large hop count, its duration can be well approximated by an exponential random variable. Moreover, the parameter of this exponential random variable can be approximated by the sum of the inverses of the expected durations of the links along the path.

1.4.2 A new path selection scheme – MED

Based on our analytical results on the path duration, we propose a novel path selection scheme that can be implemented in existing on-demand routing protocols. The *Maximum Expected Duration* (MED) path selection scheme can select the path with the largest expected duration and improve the efficiency of local path recovery.

It has been implemented in the AODV protocol and its performance gain has been compared with the original AODV protocol using NS-2 simulation.

Roughly speaking, the MED scheme works as follows. Each source tries to gather information for estimating the expected duration of a path during the path discovery phase. This is done by letting each intermediate node report to the source node its estimate of the expected duration of its downstream link towards the destination using the route reply messages. When multiple paths are discovered by the source, the one with the maximum estimated expected duration is selected as the primary path and the other paths are cached as backup ones. We will show in the Chapter 7 that the performance gain in terms of the durations of the paths selected under the MED scheme is quite significant compared to those using other selection scheme.

In addition to finding paths with larger expected durations, this scheme can also improve the efficiency of local path recovery. When a path failure occurs, instead of broadcasting route request messages within a local neighborhood of the broken link as most protocols do, under the MED scheme, nodes attempt to use cached paths to resume traffic. Using the MED scheme, nodes can estimate the probability that a cached path is still available. If this probability exceeds a chosen threshold, the node will start using this cached path immediately. Thus, it can reduce the delay during path recovery, and avoid using cached paths that are not likely to be available.

1.5 Outline of the dissertation

The rest of the dissertation is organized as follows: In Chapter 2, we review existing routing protocols for MANETs and discuss their path selection criteria. There are in general three categories of routing protocols: proactive, on-demand and hybrid protocols.

In Chapter 3, we describe the model and introduce a parametric scenario used to study the distribution of path duration. A set of modelling assumptions are also discussed. In Chapters 4 and 5, we present the distributional convergence of path duration as the path hop count increases.

In Chapter 6, we propose our new path selection scheme, MED. Its implementation in AODV is described in detail. In Chapter 7, the performance gain using the new path selection scheme is presented. The comparison between our new protocol and some of the existing protocols suggests significant benefit of using the MED path selection scheme. We also show some simulation results that are used to justify the assumptions imposed in Chapters 4 and 5.

In Chapter 8, we conclude the thesis and identify directions for future research.

Chapter 2

Review of routing protocols for MANETs

In this chapter, we review some routing protocols proposed for MANETs. As mentioned in Section 1.1, we face many challenges when designing a good routing protocol for MANETs. Moreover, there are many different types of MANETs, which can be distinguished by the number of nodes participating in the network, the mobility pattern of nodes, the traffic demands of nodes and etc. Different types of MANETs have different requirements on the performance of their routing protocols. Hence, there is no single routing protocol that works for all types of networks.

Generally speaking, there are three types of routing protocols proposed for MANETs, each of them has distinct features and are suitable for certain types of network environment. The first one is called table-driven or proactive routing. The idea behind this type of routing protocol is similar to the routing protocols designed for wireline networks. The second one is called on-demand or reactive routing. One of the distinctive features of this type of routing protocol is that nodes participate in routing only when necessary. The third type is called hybrid, which combines the features from both table-driven and on-demand routing protocols. Some simulation results on the comparison of the performance of a few routing protocols to be discussed in Section 2.1 and 2.2 are available in [28].

To summarize, some common desirable features for routing protocols of MANETs include: (i) High adaptivity to topology changes, (ii) Minimal overhead, and (iii) Capability of providing certain quality of service (QoS). As applications become more sophisticated, they demand the network protocols to provide certain QoS guarantees.

Followed by the review of the routing protocols, we will discuss in Section 2.4 the path selection criteria used by these protocols and reveal a problem remains unsolved.

2.1 Table-driven routing protocol

Under table-driven routing protocols, nodes attempt to maintain routing information regarding every other node at all times. This is done by requiring each node periodically broadcast its location information. After gathering global information of the topology of the network, each node tries to calculate the "best" path to every possible destination, and stores this information in its routing table. When a connection to a destination is needed, a node consults its routing table to find a path if the destination is reachable.

Consistency of the routing information at all nodes is crucial for the routing algorithm to operate efficiently without creating loops in the network. In order to maintain up-to-date routing information for all the nodes, whenever there is a change in the network topology, *e.g.,* nodes joining or leaving the network or simply link failure happening, the change in routing information needs to be disseminated to the rest of the network.

In some MANETs, the speed at which nodes move can be very slow. If so, the route control overhead associated with routing update can be very low since network topology changes only infrequently. Besides, under a table-driven routing protocol, if a destination is reachable, initiating traffic only takes a negligible amount of time by searching through the routing table. Thus, table-driven routing protocols can perform efficiently in this case. However, if the nodes move at moderate or higher speeds, large overhead associated with routing updates may occur. Moreover, if the traffic demand is very light, then nodes waste much time in updating routing information that are not necessary for relaying their data packets. In MANETs, nodes usually operate on battery power, hence this waste of energy can significantly shorten the lifetime of a network. Therefore, table-driven routing protocols are not considered appropriate for such an environment.

We briefly review three protocols in this category: Destination Sequenced Distance Vector, Wireless Routing Protocol, and Cluster-head Switch Gateway Routing.

2.1.1 Destination Sequenced Distance Vector protocol

Destination Sequenced Distance Vector (DSDV) protocol was developed in 1994 by C. Perkins [35]. It is based on the classical distributed Bellman-Ford routing algorithm [3,11]. Each node in the network maintains a routing table that contains all known destinations and the hop distance to each of them.

For each entry of the routing table, there are five fields: the destination ID, next hop node ID, number of hops to the destination, sequence number associated with this entry and the time the entry is established. The sequence number shows the freshness of the routing information to the destination. Each time the routing information regarding a node is updated, a higher sequence number will be associated with the node.

The DSDV protocol uses the hop count to the destination as a metric when comparing two paths to the same destination if the two paths have the same sequence number. The one with the smallest hop count is selected as the primary path for data transmission when more than one path are available.

Routing tables at all nodes are updated throughout the network immediately after a significant topology change that leads to a path information change. The messages regarding routing information updates can be disseminated using either full dump or incremental dump. When a full dump is used, the information in the whole routing table at the transmitting node is broadcasted. When an incremental dump is used, only information that has changed since the last full dump is broadcasted.

During the process of route updating, fluctuations in routing table or broadcast storm can occur due to asynchronous broadcast times of different nodes. For example, a node i may receive two route control messages regarding the same destination with the same sequence number. If for some reason, the route with the worse metric arrives to node i earlier, then without paying to much attention to it, the node simply forwards this information to its neighbors. However, since the route control message with the better metric should arrive shortly and will be forwarded again, broadcasting of the first routing updates could have been saved. Therefore, in order to avoid selection of a suboptimal path, nodes wait an additional amount of time, known as a settling time, before broadcasting the updating messages.

2.1.2 Wireless Routing Protocol

Wireless Routing Protocol (WRP) [30] is a distance vector routing protocol. Under the WRP protocol, nodes exchange the hop distance and second-to-last hop (predecessor) information for each destination. Each node maintains four tables: (a) distance table, (b) routing table, (c) link-cost table, and (d) message retransmission table. The link cost table stores delay information associated with links. The path with the minimum end-to-end delay is used. The message retransmission table contains the sequence number of the update message, a retransmission counter and a list of updates sent in the update message. Under the WRP protocol, each node constantly performs consistency check of predecessor information reported by all its neighbors in order to avoid the count-to-infinity problem and to ensure route convergence.

2.1.3 Cluster-head Switch Gateway Routing

The Cluster-head Switch Gateway Routing (CSGR) [9] protocol uses the DSDV protocol as the underlying routing algorithm with a hierarchical cluster-head-to-gateway routing approach. Under the CSGR protocol nodes are grouped into clusters. Each cluster has a cluster head. The nodes that are within the transmission range of two or more cluster heads serve as gateways. Each node keeps a cluster member table that stores the cluster head information of each destination. Cluster heads are responsible for relaying traffic for the nodes in their own clusters through gateways.

When a node S wants to send a packet to a destination node D, it first looks up its routing table to find the cluster head of node D, denoted by $H(D)$. It then transmit the packet to its own cluster head. The cluster head will find out the gateway shared with the next cluster head along the route to the cluster head $H(D)$. Finally, the cluster head $H(D)$ delivers the packet to the destination. In contrast to most other distant vector protocols, the size of routing tables are much reduced at a large number of ordinary nodes, since they only need to maintain one entry for nodes in the same destination cluster. The burden of routing is now shifted to the cluster heads and gateways. Thus, single point failure becomes a problem. If cluster heads or gateways fail or change frequently, then the performance of this protocol can be significantly degraded.

Cluster heads are elected using a Least Cluster Change (LCC) clustering algorithm. This algorithm can be done in a distributed manner. Changing cluster heads will cause large routing overhead, and thus should be avoided if possible. A cluster head change occurs only when two cluster heads move into the clusters of each other, or one of the nodes moves out of the transmission ranges of all the cluster heads. Within a cluster, the cluster head has a priority for packet transmissions, because it needs to broadcast messages, forward packets and schedule the channel uses among other nodes.

2.2 On-demand routing protocol

On-demand routing protocols comprises another category of routing protocols designed for ad-hoc networks. Under these routing protocols, nodes obtain and maintain routing information for destinations only upon request. A path discovery procedure is initiated by a source node, during which the source broadcasts a path request message to its neighbors. Neighboring nodes keep re-broadcasting this request messages until it reaches either the destination or an intermediate node with known path information to the destination. When a path in use breaks down, a path recovery is triggered to find an alternative path if data traffic is still in progress.

Compared with table-driven protocols, these protocols are more desirable when the topology changes frequently while the path request arrivals are not so frequent. This is because unlike under table-driven routing protocols, nodes only process the routing information that is relevant to their own traffic demand instead of maintaining up-to-date routing information of the entire network, thus reducing routing overhead and energy consumption both for communication and computation requirements. However, since no known path may be available at the source when a path request arrives, an additional delay is incurred during the path discovery phase before data can be transmitted.

Examples of on-demand routing protocols include Dynamic Source Routing, Ad-hoc On-demand Distance Vector routing, Temporally Ordered Routing Algorithm, and Associativity-Based Routing protocol.

2.2.1 Dynamic Source Routing

The Dynamic Source Routing (DSR) protocol [21] is based on the concept of source routing. Under source routing, the sender of a packet specifies in the packet header the entire route along which the packet should take towards the destination. This is quite different from the hop-by-hop packet routing strategy. Each node maintains the source routes to all destinations it is aware of. There are two phases upon which the DSR protocol operates: route discovery phase and route maintenance phase.

When a path is requested, the source node first searches through its routing table. If no known path is cached, it initiates a route discovery phase, during which a Route Request message (RREQ) is broadcasted. Each node that receives the request message checks if it has the routing information to the requested destination. If not, it adds its own address to the request message and rebroadcasts it. A Route Reply message (RREP) is generated if it reaches either the destination or an intermediate node that has a route to the destination. The RREP message contains the entire routing information from the source to the destination. When several paths are available between a source and a destination, the path with the smallest hop count is selected.

Under the DSR protocol, every node is responsible for delivering the data packets to the next hop specified in the packet header. Each packet can be retransmitted up to some maximum number of times until a confirmation is received from the next hop. In case of a link failure and/or no confirmation is received, the node that detects the link failure generates a Route Error message (RRER), and forward it to the source. The source then removes the route from its cache. This is called route maintenance phase. No local recovery is performed. Only the source can re-initiate the path discovery if a path is still needed.

The DSR protocol has a few drawbacks. First, each data packet contains the entire routing information (*i.e.,* the IDs of nodes along the entire path) in its header. Thus, the amount of overhead per data packet is fixed. Second, because it is based on a Link-State-Algorithms that requires each node construct its own "map" of the entire network in order to find the best path to a destination, it is only efficient in MANETs with less then 200 nodes and with nodes moving at moderate speeds [49].

2.2.2 Ad-hoc On-demand Distance Vector routing

The Ad-Hoc On-Demand Distance Vector (AODV) routing protocol [36] is a reactive routing protocol that uses a distance vector algorithm. Unlike the DSR protocol, packets are forwarded hop-by-hop, *i.e.,* each node only needs to know the next hop to forward the data packets. The AODV protocol is designed for networks with tens to thousands of mobile nodes [36, 50].

During the path discovery phase, the source node broadcasts a route request packet to its neighbors. Each path request contains a request ID generated by the source. Each time a source generates a request message, it increases the request ID by one. When a node receives multiple copies of the same request message, identified by the source ID and request ID contained in the request, only the one received first is kept and the rest are discarded. Using this scheme, loops are prevented.

In the routing table at each node, there is a route entry for each known destination. Each entry contains the destination ID, sequence number of the destination, hop count to the destination, next hop ID to the destination, and time to expire. The sequence number of the destination is an indicator of the freshness of the route entry. When a node receives a request message, if it has a route to the destination, then it needs to further check if the sequence number required by the source is less than the sequence number of the destination in the entry. If so, a reply message is generated. Otherwise, the node rebroadcasts this route request message.

When a link failure occurs, the node that detects the link failure tries to perform a local recovery. It broadcasts a path request message to its local neighborhood to inquire if any neighboring node has an alternative path to the destination. If a local recovery is unsuccessful, then the node will generate a route error message, and forward it to the source. The source will then re-initiate a path discovery phase.

2.2.3 Temporally Ordered Routing Algorithm

The Temporally Ordered Routing Algorithm (TORA) protocol [33] is proposed to operate in a highly dynamic mobile wireless environment. It is based on the ideas from a link-reversal routing protocol, Gafni-Bertsekas, [12]. Under the TORA protocol, each node maintains an ordered quintuple as its "height" metric. The height of a node is defined by a reference level and an offset with respect to that reference level.

When a path is requested, nodes use a conceptual "height" metric to establish a directed acyclic graph rooted at the destination. A downstream link is established from a node at a higher "height" to a node at a lower "height". Multiple routes can be established in parallel for a source-destination pair. When a link fails, nodes change their reference levels accordingly so that the link is reversed and previous path becomes invalid. This is where the "link reversal" notion comes from.

Since multiple paths are used, the source does not need to initiate a new path discovery phase as long as one of the routes remains valid. The key advantage of the TORA protocol is that it limits the propagation of topology change information to a local neighborhood where the link fails. However, the consistency of the height metrics among nodes relies on the temporal order of the topological changes. Therefore, implementing the TORA protocol in a real ad-hoc network requires nodes to synchronize by accessing an external time source such as the Global Position System.

2.2.4 Associativity-Based Routing

The Associativity-Based Routing (ABR) protocol attempts to provide a path that is likely to be more stable than other available paths through a path selection step during the route discovery and route recovery phase. When a source needs to establish a route, it broadcasts a query. Each intermediate node places a measure called associativity into the packet header and rebroadcasts the query until it reaches the destination. The so-called associativity measures a level of stability of a link [43] as follows: Each node constantly scans for the beacons transmitted from its neighboring nodes and records the number of beacons it receives from each of them. This number is called the node's associativity with a particular neighbor. The underlying assumption for using this associativity measure is that the larger the number of the recorded beacons is, the less mobile the node is [43]. Hence the path should be established using these stable links. Alternatively, the associativity can be viewed as an approximation of the age of a link.

The destination may receive a broadcast query through multiple paths. When this happens, it chooses the path that has the maximum average associativity (*i.e.,* the aggregated associativity divided by the hop count). The destination sends a reply packet along the selected path. The other discovered routes are then discarded.

When any link along the path breaks down because of node mobility, a local recovery of the route is attempted first before the source rebroadcasts another route query. The intermediate node that detects the link failure broadcasts a query to discover an alternative route. The destination also performs the path selection when it receives this query.

Although the ABR protocol attempts to take into account link level information when selecting a more stable path, there are a few shortcomings. First, the ABR protocol requires the destination node to select the path. Unlike most other on-demand routing protocols under which any node aware of a route to the destination can generate a reply message to the source, the route query has to reach the destination before a route can be selected. Therefore, it incurs higher delays. Moreover, a local recovery is efficient only when the link failure is close to the destination.

Second, the destination selects the most "stable" route based on the average associativity of the path. However, since a selected path remains available only until one of the links goes down, it is the residual life of that link that determines the path duration, not the average "age" of the link.

2.3 Hybrid routing protocol

A hybrid routing protocol combines the features of table-driven and on-demand routing protocols. It maintains proactive routing locally, but uses reactive routing globally.

An example of hybrid protocol is Zone Routing Protocol (ZRP) [17]. Each node specifies a quantity called a zone radius, which is measured in hops. A node's routing zone is defined as a collection of nodes whose minimum hop distance from the node in question is less than its zone radius. The ZRP protocol uses table-driven routing algorithms for forwarding packets within a zone and on-demand routing algorithms for inter-zone communications. These protocols have attractive features in terms of overhead saving compared with purely table-driven routing protocols. However, as intra-zone communication and inter-zone communication rely on two different independent routing protocols, the stability of the routing problem poses a potential problem.

2.4 Review of existing path selection criteria

When multiple paths exist between a source destination pair, the source usually selects one of them as the primary path for data transmission. If multipath routing is supported, then the source can select more than one path and use them in parallel. In either one of the cases, the source needs to adopt some path selection criterion. This criterion is usually driven by the QoS requirement of the traffic. If the QoS requirement concerns the delay of the packets, then the path with minimum delay should be chosen. Similarly, if the QoS requirement is to ensure certain bandwidth or to limit the packet drop probability, then the path that meets these requirements is selected, and etc. Path selection based on these criterions are discussed in [1, 23, 45]. However, the underlying assumptions in these criteria is that all paths are more or less stable so that ongoing network services will not likely to be interrupted due to path failure caused by mobility of the nodes.

Unlike traditional wireline networks, nodes in MANETs may move constantly at moderate or higher speeds. Network services are also likely to be interrupted due to the mobility of noise, besides other negative effects such as interference, channel access collision or even power failure of nodes. Therefore, to ensure quality of service in such a dynamic environment, we incorporate the expected duration of paths into the criteria of path selection.

The most widely adopted path selection criterion is based on the path hop count. Minimum-hop path is often considered the best solution when no other QoS requirement is enforced. This is a reasonable approach for a wireline network, since usually the routing overhead, processing and transmission delays increase with increasing hop counts along the paths. However, in MANETs, the path hop count does not directly affect the distribution of path durations. This will become clear from discussions in Chapters 4 and 5.

Moreover, if some of the nodes move at relatively high speeds, while others at moderate or slow speeds, then intuitively the durations of paths using slower moving nodes will be longer than those using fast moving nodes with high probability. However, using path selection criteria either based on the path hop count (such as done under the DSR protocol), or the order the path is discovered (such as done under the AODV protocol), the mobility of nodes are not distinguished and thus ignored. Therefore, these are not suitable criteria for path selection in MANETs.

Under the ABR routing protocol, the metric called associativity is used to measure the stability of a path. As discussed at the end of Section 2.2.4, it is not the right approach for selecting paths with longer expected durations either.

Chapter 3

Model and assumptions

In this chapter, we describe the framework used to study the statistical properties of path durations in large scale MANETs. Definitions and modelling assumptions are introduced.

The notation and convention in this dissertation are summarized here: All random variables (rvs) are defined on a common probability space $(\Omega, \mathcal{F}, \mathbf{P})$. Two rvs X and Y are said to be *equal in law* if they have the same distribution, which is denoted by $X =_{st} Y$. The independence between two rvs X and Y is denoted by $X \perp Y$. Let $\mathbf{m}(G)$ denote the first moment of a random variable whose cumulative distribution function (CDF) is given by G. We assume $m(G) < \infty$. Convergence in distribution as n goes to infinity is denoted by $\Longrightarrow_n$. For any $\boldsymbol{x}$ in $\mathbb{R}^2$, with components (x_1, x_2), set the norm of $\boldsymbol{x}$ to be $||\mathbf{x}|| = \sqrt{x_1^2 + x_2^2}$.

3.1 A basic framework

We are interested in studying the distribution of path duration in a large scale MANET, large in both the number of the nodes and the domain of the network. Let N, $N \in \{1, 2, \dots\}$, denote the number of mobile nodes in the network and let $V = \{1, \dots, N\}$ denote the set of nodes. These nodes move across a domain $\mathcal{D}$ of $\mathcal{R}^2$ or $\mathcal{R}^3$ according to certain mobility models. The routing protocol used is assumed to be an on-demand one.

3.1.1 Reachability processes

A link between node i and node j (different from node i), denoted by (i, j), is established as soon as these two nodes become reachable. The status of the link (i, j) is either up or down. Thus, we can model the status of the link using a $\{0, 1\}$-valued reachability process $\{\xi_{ij}(t), t \geq 0\}$. The interpretation of the reachability process is that $\xi_{ij}(t) = 1$ (resp. $\xi_{ij}(t) = 0$) if the link between node i and node j is up (resp. down) at time t, $t \geq 0$. It is obvious that the reachability process is a simple alternating on-off process. The duration of successive up and down times of link (i, j) are given by the rvs $\{U_{ij}(k), k = 1, 2, \dots\}$ and $\{D_{ij}(k), k = 1, 2, \dots\}$, respectively, starting from time 0, if link (i, j) is up at time 0.

There are many ways to define the reachability of two nodes. The definition can be chosen based on the purpose of study. For example, if the connectivity of two nodes is only based on their geographic locations, then we can define two nodes to be reachable if the distance between them is less than their common transmission range. If we intend to consider some physical layer characteristics of the channels, then two nodes can be considered reachable if the signal to interference and noise ratio (SINR) at the receiver exceeds a certain threshold, and packets from each other can be decoded successfully.

We introduce four commonly used definitions of this link connectivity in the area of wireless communication. The first two models are based on the locations of nodes. In a disk model [14, 37], a link set up by two nodes is considered to be up if the distance between the two nodes is smaller than certain threshold value $r_{min} > 0$. Usually this threshold is set to be the minimum of the transmission ranges of the pair of nodes. Using the notation defined earlier, we have

$$\xi_{ij}(t) := \mathbf{1}\left[||\boldsymbol{X}_i(t) - \boldsymbol{X}_j(t)|| \leq r_{min}\right], \quad t \geq 0, \tag{3.1}$$

where, for each i in V, $\boldsymbol{X}_i(t)$ denotes the position of node i at time $t \geq 0$. Under this model, transmissions between these two nodes are assumed to be successful if (3.1) holds. However, it does not include the effect of other simultaneous transmissions in the network that cause interference. The other model is called Protocol model [14]. Under this model, transmission from node i to node j is considered successful (link (i, j) is up), if the distance between node j and any other transmitting node k is larger than the distance between node i and node j by a factor of $(1 + \Delta)$, where $\Delta > 0$. This can be written as

$$\xi_{ij}(t) := \mathbf{1}\left[||\boldsymbol{X}_k(t) - \boldsymbol{X}_j(t)|| \geq (1 + \Delta)\, ||\boldsymbol{X}_i(t) - \boldsymbol{X}_j(t)||\, \forall k \in V\right], \quad t \geq 0\,. \tag{3.2}$$

In this model, so as the SINR model to be introduced shortly, a link may not be bidirectional due to different interference each node experiences.

The next two models take into account more physical layer characteristics of the channels. Both can be written in the following form: Two nodes i and j in V can maintain a link between them at time $t \geq 0$ if and only if

$$\min\left(\frac{P_j \cdot F_{ji}(t)}{\Psi_i(t)}, \frac{P_i \cdot F_{ij}(t)}{\Psi_j(t)}\right) > \Gamma \tag{3.3}$$

for some threshold $\Gamma > 0$, where P_i is the maximum transmission power of node i, and $\boldsymbol{F}(t) = (F_{ij}(t))$ denotes the channel gain matrix (including fading) at time t with $F_{ji}(t) \geq 0$ and $F_{ii}(t) = 0$, $i, j = 1, \dots N$.

The choices of $\Psi_i(t)$ in (3.3) depends on the physical layer characteristics we intend to model. For example, if we assume that packets received at node i from node j can be decoded successfully if and only if the received signal power at node i exceeds the threshold $\Gamma > 0$ [8,39], then we can let $\Psi_i(t) = 1$. The term $\frac{P_j \cdot F_{ji}(t)}{\Psi_i(t)}$ in (3.3) then becomes the maximum achievable received signal power at node i. Similarly, if we assume successful decoding of packets at node i requires that the achieved SINR exceeds the threshold Γ [14], then we set

$$\Psi_i(t) = W_i + \sum_{k \in TX(t) \setminus \{j\}} P_k(t) \cdot F_{ki}(t) \ , \tag{3.4}$$

where W_i is the noise power at node i, $TX(t)$ is the set of transmitters at time t and $P_k(t)$ denotes the transmission power of node k. It is clear that $\Psi_i(t)$ now represents the sum of noise power and interference at node i at time t. Thus, the definition in (3.3) simply reads that nodes i and j are reachable if and only if the achieved SINR value using the maximum transmission power exceeds Γ in both directions. This is called physical model or SINR model in [14].

3.1.2 Path duration

We assume that links defined or used in this dissertation are bidirectional. This is because reliable communication usually requires acknowledgments from downstream nodes for each transmission. Under this assumption, we can allow packets to travel along the same path from the source to destination and vice versa. However, this bidirectional link assumption is not necessary for establishing our analytical results in Chapters 4 and 5.

Having defined the set of nodes and their connectivity model, we introduce a time-varying graph, denoted by $(V, E(t))$. The set of vertices V is composed by the set of nodes in the network. The set of edges $E(t)$ consists of undirected edges through the relation

$$E(t) := \{(i, j) \in V \times V : \ \xi_{ij}(t) = 1\}, \quad t \geq 0 \ . \tag{3.5}$$

By convention, we set $\xi_{ii}(t) = 0$ for each $i \in V$ and all $t \geq 0$. Thus, a path can be established between nodes s and d at time $t \geq 0$, if node d (s) is reachable from node s (d) through a set of edges in the undirected graph $(V, E(t))$. The set of paths from node s to node d is denoted by $\mathcal{P}_{sd}(t) \subseteq 2^{E(t)}$. It can be empty if there is no path between nodes s and d at time t. It can also contain multiple paths that may exist between nodes s and d. In such a case, let $\mathcal{L}_{sd}(t)$ denote the set of links along the path selected by the routing protocol.

An established path between a source destination pair is considered invalid as soon as one of its links breaks down. Thus, the *duration* of a path is defined to be the amount of time that elapses starting from the path set-up time until one of the links along the path goes down. To simplify our analysis, we ignore the path set up delays.

For each link ℓ, let $T_\ell(t)$ denote the amount of the time that elapses from time t onward until link ℓ goes down. We shall also call it the time-to-live or excess life of link ℓ after time t. For a path set up at time t, its duration is determined by the minimum of the excess lives after time t of the links along the path. Let $Z_{sd}(t)$ denote the duration of a path from node s to node d established at time t. The set of links along the path is denoted by $\mathcal{L}_{sd}(t)$ as already mentioned. Summarizing what we have defined, the path duration $Z_{sd}(t)$ can be written as

$$Z_{sd}(t) := \min\left(T_\ell(t) : \ell \in \mathcal{L}_{sd}(t)\right), \quad t \geq 0 . \tag{3.6}$$

3.2 The set-up and modelling assumptions

In this dissertation, we focus on studying the distribution of path durations between randomly selected pairs of source and destination nodes in large scale MANETs. As we will explain in the following, this problem can be translated into studying the distribution of path duration for paths with large hop count. For a fixed number of nodes, uniformly distributed in the domain $\mathcal{D}$, the geographical distance between a pair of randomly selected nodes is expected to increase as the size of the domain increases. If the transmission ranges of the nodes remain fixed, then we need to add more nodes into the network work in order to obtain a connected graph. Thus, for a pair of randomly selected source and destination that are reachable from each other through a multi-hop path, their hop distance will increase as the network size increases. The size of a network refers to both the network domain and the number of nodes.

In order to define how the network scales, we introduce a parametric scenario that consists of a sequence of networks denoted by the pair $(V^{(n)}, \mathcal{D}^{(n)})$, $n = 1, 2, \ldots$. For each n, let $V^{(n)} = \{1, \ldots, N^{(n)}\}$ denote the set of mobile nodes and $\mathcal{D}^{(n)}$ the domain across which the nodes move. For each node i in $V^{(n)}$, let $\boldsymbol{X}_i^{(n)}(t)$ denote the location of node i in domain $\mathcal{D}^{(n)}$ at time t. We call the process $\{\boldsymbol{X}_i^{(n)}(t),\ t \geq 0\}$ the trajectory process of node i in $\mathcal{D}^{(n)}$. The stochastic process that governs the arrival of path requests is assumed to be independent of these trajectory processes.

3.2.1 Scaling

We take a simple form of scaling as follows: Let

$$N^{(n)} \sim nN^{(1)} \quad \text{and} \quad \text{Area}(\mathcal{D}^{(n)}) \sim n \cdot \text{Area}(\mathcal{D}^{(1)}) \tag{3.7}$$

as n goes to infinity [1]. The relation $N^{(n)} \sim nN^{(1)}$ should be understood as $\lim_{n\to\infty} \frac{N^{(n)}}{nN^{(1)}} = 1$. Under this scaling, we have

$$\frac{N^{(n)}}{Area(\mathcal{D}^{(n)})} \sim \frac{N^{(1)}}{Area(\mathcal{D}^{(1)})} .$$

In other words, the density of nodes or the number of nodes per unit area remains the same. For our convenience, we reparameterize $N^{(n)}$ so that $N^{(n)} = n$.

3.2.2 Stationarity

As the network is expected to run for a long time, we assume that the network reaches its steady state. This is modeled by taking the $N^{(n)}$ trajectory processes to be *jointly stationary*. It then follows that the $\frac{N^{(n)}\times(N^{(n)}-1)}{2}$ reachability processes are jointly stationary.

As defined in Section 3.1.1, the reachability process between a pair of nodes is an on-off alternating process. If the link $\ell = (i,j)$ is on at time 0, then starting from time 0, the successive up and down times can be written as a sequence $\{U_\ell^{(n)}(1), D_\ell^{(n)}(1), U_\ell^{(n)}(2), D_\ell^{(n)}(2), \dots\}$. Since we assume that the network has reached its steady state, we then necessarily have these reachability processes as on-off alternating processes in equilibrium.

To study a process in equilibrium, we can start the process at time 0, and observe it at time infinity. Alternatively, as we will do here, we can view the processes starts from time $-\infty$, and observe it from time 0.

[1] From now on we omit this qualifier in all asymptotic equivalences.

Some of well known results on alternating renewal processes in equilibrium are reviewed in Section 3.2.2.1. Please see [10, 18] for more details. We will continue with the stationarity assumption applied in the context of our problem in Section 3.2.2.2.

3.2.2.1 Review on renewal process

The formal definition of a renewal process is the following [18, p.109]:

Definition 3.1 *Let* X_i*,* $i = 1, 2, \ldots,$ *be nonnegative i.i.d random variables and* $S_n = \sum_{i=1}^{n} X_i$*,* $n = 1, 2, \ldots.$ *The renewal process* $\{N(t), t \geq 0\}$ *is defined by*

$$N(t) = \sup\{n : S_n \leq t\}, t \geq 0 \ .$$

The process $N(t)$ is a counting process, which counts the number of renewal epochs from time 0 to time t. The X_i's are the inter-arrival times of two successive renewal epochs. If the time to the first renewal epoch, X_1, has a different distribution from that of other X_i's, $i = 2, 3, \ldots,$ then $N(t)$ is called a *delayed renewal process.*

Two important random variables associated with a renewal process are its forward and backward recurrence times. The backward recurrence time is defined by

$$A_t = t - S_{N(t)}, \ t \geq 0 \ .$$

It represents the amount of time since the last renewal before time t. Similarly, the forward recurrence time that represents the amount of time from t until the next renewal occurs is defined as

$$B_t = S_{N(t)+1} - t,\ t \geq 0 \ .$$

This forward recurrence time is also called the excess or the residual life at time t. The forward and backward recurrence times have the same CDF function, $F(\cdot)$, given by

$$F(z) = \frac{1}{\mathbf{E}[X]} \int_0^z (1 - G(x))\ dx \ , \tag{3.8}$$

where X denotes a non-negative rv that has the same support and distribution as X_i's, and $G(\cdot)$ denotes the CDF of X. In general, we call a rv the forward recurrence time associated with X if it has a CDF defined in (3.8).

Having defined the forward and backward recurrence times, we can define the renewal process in equilibrium.

Definition 3.2 *The renewal process $N(t), t \geq 0$, is in equilibrium if it is a delayed renewal process with X_1 distributed according to the forward recurrence time associated with X.*

We are now ready to define the equilibrium alternating renewal process. A realization of an alternating renewal process is shown in Fig. 3.1. In this example, the process is assumed to start from "on" status at time 0. Then, at its first renewal epoch, $t = X_1$, it goes to the "down" status. It goes back to "on" status at time $t = X_1 + Y_1$, and so on. The random variables, $X_i, i = 1, 2, \ldots$ are *i.i.d* with generic marginal X. The rvs Y_i's are also *i.i.d* and independent of rv X_i's.

This alternating renewal process is said to be in equilibrium if X_1 is distributed as the forward recurrence time associated with X [10]. The process can also start from "off" status and be in equilibrium, in which case X_i's are replaced with Y_i's, *i.e.*, Y_1 is distributed as the forward recurrence time associated with Y_2.

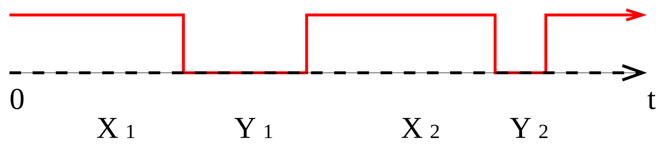


Figure 3.1: On-off process

3.2.2.2 Implication of stationarity assumption

In the context of our problem, for link $\ell = (i, j)$, we let $G_\ell^{(n)}$ denote the CDF of $U_\ell^{(n)}$. The forward recurrence time associated with rv $U_\ell^{(n)}$ has CDF $F_\ell^{(n)}(x)$ given by

$$F_\ell^{(n)}(x) = \begin{cases} \frac{1}{\mathbf{m}(G_\ell^{(n)})} \int_0^x \left(1 - G_\ell^{(n)}(y)\right) \, dy & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \,. \end{cases} \tag{3.9}$$

Under the assumption that the reachability process is in steady state, the excess live of link ℓ, $T_\ell^{(n)}(t)$ has CDF $F_\ell^{(n)}(\cdot)$.

We introduce a sequence of rvs $\{X_\ell^{(n)}(t)\}$, $\ell = 1, 2, \ldots$, and assume that

$$X_\ell^{(n)}(t) =_{st} [T_\ell^{(n)}(t) | \xi_\ell^{(n)}(t) = 1] \,,$$

for each $\ell \in \mathcal{L}_{sd}(0)$. In other words, $X_\ell^{(n)}$ models the excess life of the ℓ-th link.

Then, we can rewrite the definition of path duration defined in (3.6) as the following:

$$Z_{sd}^{(n)(t)} := \min\left(X_\ell^{(n)}(t) : \ \ell = 1, \ldots, |\mathcal{L}_{sd}^{(n)}(t)|\right) \ . \tag{3.10}$$

Under the stationarity assumption, it suffices to consider only the case when $t = 0$. For notational convenience, we will drop this time index as no confusion will arise.

It is obvious from (3.9) that the excess life of any link has a non-increasing probability density function (PDF). Moreover, the choice of link duration distribution $G_\ell^{(n)}$ can depend on link dynamics due to either nodes' mobility or channel fading.

3.2.3 Deterministic sequence of path hop count

The difficulty in studying the path duration defined in (3.10) is twofold. First, the rvs $X_\ell^{(n)}$ are dependent. This is because the $\frac{N^{(n)} \times (N^{(n)}-1)}{2}$ reachability processes are usually *not* mutually independent, even when the $N^{(n)}$ trajectory processes of the nodes are mutually independent. For example, two adjacent links along a path have dependency in their link excess lives due to their shared middle node. Although this dependency causes difficulty in our analysis, we will show in Chapter 5 the asymptotics of path durations in large scale MANETs under certain mild assumptions.

Secondly, the set of links $L_{sd}^{(n)}(0)$ is a *random* subset of $E^{(n)}(0)$. It also depends on the underlying routing protocol. However, under the parametric scenario introduced earlier and with a disk model adopted, we expect that the number of hops between a randomly selected source and destination node is going to increase approximately as $\sqrt{n}$. Hence, we assume that there exists a pair of nodes s and d in $\mathcal{V}^{(n)}$ such that $\lim_{n\to\infty} |\mathcal{L}_{sd}^{(n)}(0)| = \infty$. To simplify our analysis, we assume that the sequence $\{|\mathcal{L}_{sd}^{(n)}(0)|, n = 1, 2, \dots\}$ is deterministic.

To summarize, we now have a sequence of rvs $\{Z^{(1)}, Z^{(2)}, Z^{(3)}, \dots\}$[2] with each $Z^{(n)}$ representing the path duration in the sequence of networks parameterized by n, *i.e.*,

$$Z^{(n)} := \min\left(X_\ell^{(n)} : \ \ell = 1, \dots, H(n)\right) \tag{3.11}$$

where $H(n) = |\mathcal{L}^{(n)}(0)|$. The definition of path duration given in the form (3.11) shall be used throughout the rest of the dissertation.

[2]The subscript of sd will be omitted from now on since the source destination pair is randomly selected.

Chapter 4

Distributional convergence of path duration – Independent case

Using the model and assumptions defined in Chapter 3, we are now ready to study the distribution of path duration. In this and the next chapters, we present the main results of this dissertation. We will show that under a set of mild assumptions, the duration of a path, when scaled properly, converges to an exponential random variable as the path hop count increases. Moreover, the expected path duration can be approximated using the expected durations of the links along the path.

As discussed in Section 3.2.3, link excess lives that define the path duration in (3.11) are not mutually independent. In this chapter, we start with a simpler case where we assumed that these link excess lives are mutually independent. This assumption allows us to apply Palm's theorem to prove the distributional convergence of path duration. Its proof is simple and clearly reveals the intuition that leads to the convergence result.

Strictly speaking, this independence assumption does not hold in general, However, it may be reasonable for some mobility models. This will be discussed in Section 4.3. A more general case with dependent link excess lives will be studied in Chapter 5.

4.1 Independence assumption

We first state the independence assumption using the notation defined in Chapter 3. The path duration, denoted by $Z^{(n)}$, is defined in (3.11), *i.e.,*

$$Z^{(n)} = \min\{X_\ell^{(n)}, \ell = 1, \dots, H(n)\} \ , \tag{4.1}$$

where $X_\ell^{(n)}$ denotes the excess life of the ℓ-th link along the path. The sequence $\{H(n), n = 1, 2, \dots\}$ has been assumed to be deterministic. Thus, the difficulty in studying the path duration is now solely due to the dependency among $X_\ell^{(n)}$'s. We now assume that these rvs $X_\ell^{(n)}, \ell = 1, \dots, H(n)$, are mutually independent.

4.2 Distributional convergence of path duration

The rv $X_\ell^{(n)}$ in (4.1) represents the excess life of the ℓ-th link. From the discussion on a renewal process in equilibrium in Section 3.2.2, we can see that the CDF of rv $X_\ell^{(n)}$, denoted by $F_\ell^{(n)}$, is related to some link duration CDF $G_\ell^{(n)}$ through

(3.9). The distributions $G_\ell^{(n)}, \ell = 1, \ldots, H(n)$, are not necessarily identical. For each $G_\ell^{(n)}$, we let $\lambda_\ell^{(n)}$ denote the inverse of its mean, *i.e.*,

$$\lambda_\ell^{(n)} := (m(G_\ell^{(n)}))^{-1}, \quad \ell = 1, \ldots, H(n),\ n = 1, 2, \ldots. \tag{4.2}$$

We are now dealing with an array of $\mathbb{R}_+$ rvs $\{X_\ell^{(n)}, \ell = 1, \ldots, H(n); n = 1, 2, \ldots\}$.

The following conditions are assumed to hold:

Assumption 4.1 *There exists $\lambda > 0$ such that*

$$\lim_{n\to\infty} \frac{1}{H(n)} \sum_{\ell=1}^{H(n)} \lambda_\ell^{(n)} = \lambda. \tag{4.3}$$

Assumption 4.2 *For every $x \geq 0$,*

$$\lim_{n\to\infty} \left(\max_{\ell=1,\ldots,H(n)} G_\ell^{(n)}(\frac{x}{H(n)}) \right) = 0. \tag{4.4}$$

A couple of remarks on these two assumptions: Assumption 4.1 is a scaling condition that allows us to define the parameter of the limiting distribution of a properly scaled path duration. Assumption 4.2 can be understood as the following: For every $x \geq 0$ and any given $\varepsilon > 0$, there exists an integer $n^\star = n^\star(x;\varepsilon)$ such that

$$\max_{\ell=1,\ldots,H(n)} G_\ell^{(n)}(\frac{x}{H(n)}) \leq \varepsilon, \quad n = n^\star, n^\star + 1, \ldots \tag{4.5}$$

If we model the link excess life as a continuous rv, then this condition holds trivially.

Our main result in this chapter is the following:

Theorem 4.1 *Under Assumptions 4.1 and 4.2, it holds that $H(n) \cdot Z^{(n)} \Longrightarrow_n \mathcal{E}_\lambda$, i.e.,*

$$\lim_{n\to\infty} \mathbf{P}\left[H(n)\cdot Z^{(n)} \le x\right] = \begin{cases} 1 - e^{-\lambda x} & \text{if } x > 0 \\ 0 & \text{if } x \le 0 \end{cases} . \tag{4.6}$$

Proof: This convergence result follows directly from Palm's Theorem [18, Thm. 5-14, p. 157]. Consider an array of renewal processes $\{N_{j,m}(t), t > 0, m = 1, 2, \ldots, j = 1, \ldots, m\}$, all in equilibrium. For a fixed m, let $N_{j,m}(t)$, $t > 0$, denote the j-th one, $j = 1, \ldots, m$. Let X_{1jm} be the epoch of the first renewal starting from time 0 and X_{2jm} be the time between the first and second renewal epochs for the j-th renewal process. Let $G_{jm}(t) = \mathbf{P}[X_{2jm} \le t]$. The renewal rate of the j-th renewal process is denoted by $\lambda_{jm} = (\mathbf{E}[X_{2jm}])^{-1}$, $j = 1, \ldots, m$.

For each m, let $N_{0,m}(t)$ be an superposition of m equilibrium renewal processes, including $N_{1,m}(t)$ to $N_{m,m}(t)$, *i.e.*,

$$N_{0,m}(t) = N_{1,m}(t) + \cdots + N_{m,m}(t), t \ge 0.$$

Palm's theorem states that under the following two assumptions, the amount of time starting from 0 until the first renewal epoch of the superposed counting process $N_{0,m}(t)$ has an exponential distributed as m goes to infinity.

Assumption 4.3 *For all m sufficiently large, there is a $\tilde{\lambda} < \infty$ such that*

$$\sum_{j=1}^{m} \lambda_{jm} = \tilde{\lambda} . \tag{4.7}$$

Assumption 4.4 *Given $\epsilon > 0$, for each $t > 0$ and for all sufficiently large m, we have*

$$G_{jm}(t) \leq \epsilon \tag{4.8}$$

for $j = 1, \ldots, m$.

Theorem 4.2 Palm's Theorem: *Under Assumptions 4.3 and 4.4, it holds that*

$$\lim_{m\to\infty} \mathbf{P}\left[N_{0,m}(t) = 0\right] = e^{-\tilde{\lambda}t} \ . \tag{4.9}$$

Since the event $\{N_{0,m}(t) = 0\}$ indicates that there is no renewal epoch for the process $N_{0m}(t)$ until time t, it follows that (4.9) is equivalent to

$$\lim_{m\to\infty} \mathbf{P}\left[\min_{j=1,\ldots,m} X_{1jm} > t\right] = e^{-\tilde{\lambda}t} \ . \tag{4.10}$$

In our problem, for each n, we have a sequence of rvs $\{X_\ell^{(n)}, \ell = 1, \ldots, H(n)\}$ that models the excess lives of the links along a path of $H(n)$ hops. For each $X_\ell^{(n)}$, there is a corresponding distribution $G_\ell^{(n)}(\cdot)$ that models the distribution of the duration of the ℓ-th link. We construct another sequence $\{\hat{X}_\ell^{(n)}, \ell = 1, \ldots, H(n)\}$ such that $\hat{X}_\ell^{(n)} = H(n)X_\ell^{(n)}$, for all $\ell = 1, \ldots, H(n)$, with corresponding $\hat{G}_\ell^{(n)}(\cdot)$. It is clear that

$$\hat{G}_\ell^{(n)}(t) = G_\ell^{(n)}(\frac{t}{H(n)}), \ t \geq 0, \ell = 1, \ldots, H(n) \ .$$

As $n \to \infty$, we have $\frac{t}{H(n)} \to 0$ for any fixed t. Under Assumption 4.2, the condition given in Assumption 4.4 is satisfied for sequence of $\{\hat{G}_\ell^{(n)}\}$. We also have the relation that $m(\hat{G}_\ell^{(n)}) = H(n)m(G_\ell^{(n)})$. Under Assumption 4.1, we have

$$\begin{aligned}\lim_{n\to\infty} \sum_{\ell=1}^{H(n)} (m(\hat{G}_\ell^{(n)}))^{-1} &= \lim_{n\to\infty} \frac{1}{H(n)} \sum_{\ell=1}^{H(n)} (m(G_\ell^{(n)}))^{-1} \\ &= \lambda\ .\end{aligned}$$

Hence Assumption 4.3 is also satisfied for array $\{m(\hat{G}_\ell^{(n)}), \ell = 1, \ldots, H(n), n = 1, 2, \ldots\}$.

Therefore, using Palm's theorem, we have

$$\begin{aligned}\lim_{n\to\infty} \mathbf{P}\left[\min_{\ell=1,\ldots,H(n)} \hat{X}_\ell^{(n)} > t\right] &= \lim_{n\to\infty} \mathbf{P}\left[\min_{\ell=1,\ldots,H(n)} X_\ell^{(n)} > \frac{t}{H(n)}\right] \\ &= \lim_{n\to\infty} \mathbf{P}\left[H(n)\cdot Z^{(n)} > t\right] \\ &= e^{-\lambda t}.\end{aligned}$$

This completes the proof. ■

Theorem 4.1 says that under a set of mild assumptions, as the path hop count increases, the path duration scaled by the path hop count converges to an exponential rv in distribution. Since rv $Z^{(n)}$ tends to decrease with increasing $H(n)$, in order to keep $Z^{(n)}$ from converging to a constant rv with value zero, it is scaled by the hop count $H(n)$. Moreover, the parameter of this limiting exponential rv is given by λ in Assumption 4.1. Thus, in a real network, for a path with sufficiently large hop count between a randomly selected source and destination pair, its duration can be well approximated by an exponential rv with parameter α, where α is the summation of the inverses of the expected durations of the links along the path.

4.3 Comments on the independence assumption

Before we end this chapter, we discuss on the independence assumption made on the excess lives of the links, $X_\ell^{(n)}$. This discussion will eventually motivate us to consider relaxing this assumption.

Due to the dependency between the reachability processes[1] $\{\xi_{ij}^{(n)}(t),\ t \geq 0\}$ $(i < j \in V)$, the link excess lives are not *independent*. The question is how strong this dependency is.

We study this dependency using simulations run with nodes of different mobility models. In particular, we compare here the dependency between link excess lives in a network where nodes move according to the Random Waypoint (RWP) mobility model and the Reference Point Group (RPG) mobility model, respectively. Please refer to Appendix A.2 and A.3 for detailed descriptions of these two mobility models.

4.3.1 Correlation coefficient between link durations

The dependency between link excess lives is measured by their correlation coefficients. For a positive integer h, we collect the excess lives of pairs of links separated by h hops along a path. Then we examine their correlation coefficients. A small correlation coefficient suggests weak dependency between the pair of links.

[1]Two adjacent links that share the common intermediate node certainly have their excess lives coupled through the common node.

The simulation is run using NS-2 network simulator. For more information on NS-2 simulator, please refer to [46, 51]. The domain of the network simulated is a two dimensional rectangular region of 2 km $\times$ 2 km. There are 200 nodes moving across this region. The AODV routing protocol is used. Nodes' transmission are fixed at 250 m. For the RWP model, the speed range of nodes is $[1, 30]$ m/s. For the RPG model, the speed range of nodes is $[1, 10]$ m/s.

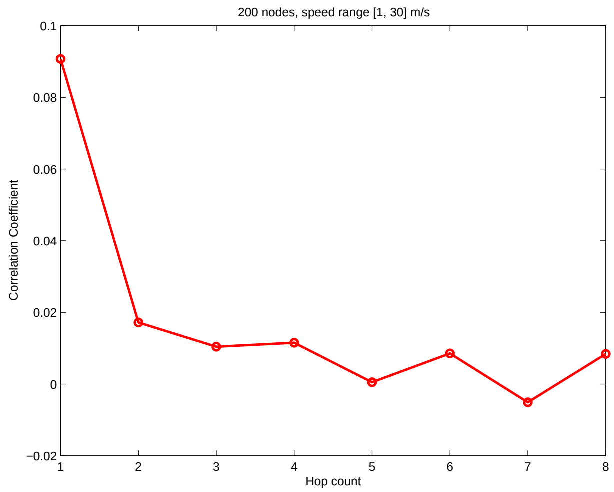


Figure 4.1: Plot of hop vs. correlation coefficient

The empirical correlation coefficients between pairs of links between two links along a path with varying hop distance under the RWP mobility model is plotted in Figure 4.1. The hop distance is taken to be the number of intermediate links between the links plus one. Hence, neighboring links have a hop distance of 1. As shown in Figure 4.1, under the RWP mobility model, the correlation coefficient between two links decreases quickly as their hop distance increases. If fact, other than between the neighboring links, the correlation coefficient is negligible. Overall, the dependency between link excess lives is not strong. Hence, the independence assumption may deemed to be reasonable in this case.

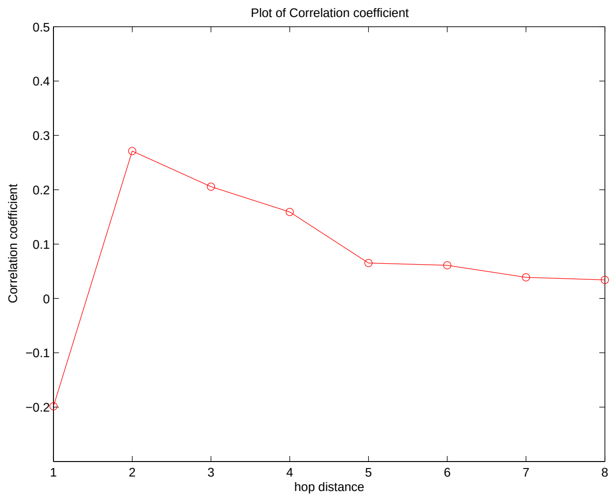


Figure 4.2: Plot of hop vs. correlation coefficient

Things are different when we consider the RPG model. In Figure 4.2[2], the absolute value of the correlation coefficient between two neighboring links can be as high as 0.26. Moreover, the correlation coefficient decreases much slower with respect to hop distance compared with what we observed under the RWP model; it still shows non-negligible correlation between two links separated by 4 or 5 hops.

Comparing these simulation results, we find that although the independence assumption may be reasonable in some cases, such as under the RWP mobility model, it may not be so under some other mobility models, *e.g.,* the RPG model. However, the simulation study in [40] suggests the same distributional convergence of path duration under those mobility models. Therefore, it is of much interest

[2]This figure is quoted from [46, Fig. 6.20] with the author's permission.

to study the distribution of path duration without imposing this independence assumption. We will show in Chapter 5 that the independence assumption is indeed not necessary for the convergence.

Chapter 5

Distributional convergence of path duration – Dependent case

In the previous chapter, we proved the distributional convergence of path duration as the path hop count increases. However, the result was established under the assumption that the excess lives of the links along the path are mutually independent. As discussed in Section 4.3, this assumption does not hold in general. The dependence between links remains non-negligible even with hop distance greater than 4 or 5 hops (*e.g.,* shown in Figure 4.2).

Motivated by this observation, we now focus on the study of the same convergence result with possible dependence among link excess lives. In Section 5.1, we first consider a special case in which the dependence of the links is assumed to be limited to a finite neighborhood. In other words, the excess lives of links that are separated by certain hop distance are assumed to be independent. In Section 5.2, we introduce a more relaxed assumption that only requires the dependence between link excess lives go away asymptotically (The exact form of this dependence decaying is defined in Section 5.2.1). In the latter case, we allow dependency in a neighborhood that is *unbounded*.

In both cases, we are able to establish the same distributional convergence in the path duration shown in Chapter 4. Furthermore, the relation between path duration and the link level statistics suggests a simple approach for the path selection scheme.

We now begin with the discussion on the finite dependence case.

5.1 Finite dependence with homogeneous link durations

In this section we consider a simpler case where the dependence between link excess lives is limited within a finite neighborhood. Furthermore, the link excess lives are modelled using a *stationary* sequence $\{X_\ell, \ell = 1, \ldots,\}$, for $\ell = 1, \ldots, H(n)$ and $n = 1, 2, \ldots,$. We let $X_\ell^{(n)} = X_\ell$, for all n. Under this assumption, the excess lives of the links along a path necessarily have the same distribution. (This is why we call it a homogenous link duration case.) Again, for all ℓ, the CDF of X_ℓ is denoted by F and is associated with a link duration CDF G in the way given by

(3.9).

The following assumption describes a finite dependence among link excess lives.

Assumption 5.1 *(m-dependence [44]) The sequence of random variables* $\{X_\ell, \ell = 1, 2, \ldots\}$, $n = 1, 2, \ldots,$ *satisfies*

$$X_\ell \perp X_{\ell'} \quad \textit{if} \quad |\ell - \ell'| > m \ ,$$

where m is a finite positive integer.

This m-dependence assumption may be reasonable as suggested by the findings in Section 4.3.1. Although the dependency between link excess lives may exist in a large neighborhood, *e.g.,* under the RPG model, we still assume the neighborhood to be bounded.

The next assumption has the same role and interpretation as Assumption 4.2.

Assumption 5.1 *For every $x \geq 0$,*

$$\lim_{n\to\infty} G(\frac{x}{H(n)}) = 0. \tag{5.1}$$

The scaling assumption for the independence case (Assumption 4.1) holds automatically here, since the distributions of links are homogeneous. Thus $G_\ell^{(n)} = G$, for all ℓ and n. Let $\lambda = (\mathbf{m}(G))^{-1}$.

We now conclude the same convergence result on the path duration.

Theorem 5.1 *Suppose that Assumptions 5.1 and 5.1 hold for the stationary sequence* $\{X_\ell^{(n)}, \ell = 1, 2, \ldots\}$ *and the CDF* G*. If the condition*

$$\begin{aligned} &\lim_{c \to 0} \frac{1}{\mathbf{P}\left[X_\ell^{(n)} < c\right]} \max_{|\ell - \ell'| \le m} \mathbf{P}\left[X_\ell^{(n)} < c, X_{\ell'}^{(n)} < c\right] \\ &= \lim_{c \to 0} \max_{|\ell - \ell'| \le m} \mathbf{P}\left[X_{\ell'}^{(n)} < c \mid X_\ell^{(n)} < c\right] \\ &= 0 \end{aligned} \tag{5.2}$$

holds, then

$$\lim_{n \to \infty} \mathbf{P}\left[H(n) \cdot Z^{(n)} \le x\right] = \begin{cases} 1 - e^{-\lambda x} \ , & \textit{if} \ \ x > 0 \\ \\ 0 \ , & \textit{if} \ \ x \le 0 \end{cases} \tag{5.3}$$

where $\lambda = (\mathbf{m}(G))^{-1}$*.*

Proof: A proof is provided in Appendix B.1 ∎

The condition in (5.2) requires that given any link with excess life smaller than c, $c > 0$, the probability that another link within hop distance m from it also has a excess life less than c goes to 0 as $c \to 0$. As $c \to 0$, the event $\{X_\ell^{(n)} < c\}$ becomes a rare event. Thus, this condition simply means that these rare events $\{X_\ell^{(n)} < c\}$ do not occur in clusters in a local neighborhood of size m.

There are different reasons that could cause this rare event. If we use a disk model for modeling the nodes' connectivity, we can interpret this rare event as the two end nodes of a link being very close to the boundaries of their transmission ranges and are about to move out of the transmission range of each other when the path is set up. For a path that is randomly chosen between a source destination pair, condition (5.2) is likely to hold. In other words, one pair of nodes being close to the edges of their transmission ranges does not necessarily imply the same for other pairs of neighboring nodes along the path. We use simulation under the RWP model and the RPG model to validate this condition in Section 7.1.1.

Theorem 5.1 again states that as the number of hops $H(n)$ along a path increases the distribution of path duration can be well approximated by an exponential distribution for all sufficiently large $H(n)$. However, Theorem 5.1 does not require the excess lives of the links along the path to be independent. It is interesting to note that the parameter of the emerging exponential distribution is the same as that for the independent case. This implies that this parameter is not affected by the dependence among link excess lives that is limited to a finite neighborhood.

5.2 General dependence with heterogeneous link duration

In the previous section we modeled the link excess lives using a stationary sequence with the same marginal distribution. The dependence between link excess lives are modeled to be limited in a finite neighborhood. In this section, we relax the assumptions on link excess lives further. First, we allow the distribution of link excess lives to be heterogeneous. Secondly, the dependency in link excess lives, instead of being limited to a finite neighborhood, decreases asymptotically with increasing hop distance.

In the next section, we define the *mixing conditions* that describe the manner in which the dependence of link excess lives decays with the hop distance between them.

5.2.1 Mixing conditions

We introduce two mixing conditions that are commonly used in the literature to describe the dependence in rvs [26, 27].

We state the mixing conditions for an array of $\mathbb{R}$-valued rvs, $\mathbf{W} := \{W_i^{(n)}, n = 1, 2, \ldots; i = 1, \ldots, h(n)\}$, where $\{h(n), n \geq 1\}$ is a sequence of positive integers with $\lim_{n\to\infty} h(n) = \infty$ and $\{u_n, n \geq 1\}$ a sequence of real numbers that also increases with n. The joint CDF of rvs $\{W_{i_1}^{(n)}, W_{i_2}^{(n)}, \ldots, W_{i_n}^{(n)}\}$ is denoted by $\mathbf{J}_{i_1\cdots i_n}^{(n)}(u_1, u_2, \ldots, u_n)$. To simplify the notation, we write $\mathbf{J}_{i_1\cdots i_n}^{(n)}(u)$ for $\mathbf{J}_{i_1\cdots i_n}^{(n)}(u, \ldots, u)$.

The first mixing condition is defined as follows:

Definition 5.1 *($D(u_n)$ condition) The array* $\mathbf{W}$ *is said to satisfy the condition* $D(u_n)$ *if for any integers*

$$1 < i_1 < \cdots < i_p < j_1 < \cdots < j_q \leq h(n) \quad \textit{with} \quad j_1 - i_p > m(n) \ ,$$

we have

$$\left| \mathbf{J}^{(n)}_{i_1 \ldots i_p j_1 \ldots j_q}(u_n) - \mathbf{J}^{(n)}_{i_1 \ldots i_p}(u_n) \mathbf{J}^{(n)}_{j_1 \ldots j_q}(u_n) \right| \leq \alpha_{n,m(n)} \ , \tag{5.4}$$

where $\lim_{n\to\infty} \alpha_{n,m(n)} = 0$ *for some sequence* $\{m(n), n = 1, 2, \ldots\}$ *such that (i)* $\lim_{n\to\infty} m(n) = \infty$ *and (ii)* $\lim_{n\to\infty} \frac{m(n)}{h(n)} = 0$.

The condition $D(u_n)$ imposes a form of "dependence decay" as the distance between two sets of rvs $\{W^{(n)}_{i_1}, \ldots, W^{(n)}_{i_p}\}$ and $\{W^{(n)}_{j_1}, \ldots, W^{(n)}_{j_q}\}$ increases. Notice that since we allow $m(n) \to \infty$, the dependency between $W^{(n)}_i$'s is not necessarily limited to a *bounded* neighborhood. This is different from the dependence of rvs under the m-dependence condition given in Assumption 5.1. In fact, any m-dependence sequence satisfies condition $D(u_n)$ with arbitrary sequence $\{u_n, n \geq 1\}$. The interpretation and role of this condition in our problem will be given shortly.

More notation and definitions are needed to state the second mixing condition. We shall call a set of consecutive integers an interval. For example, $I = \{1, 2, \ldots, 10\}$ is an interval of length ten. Let k be an positive integer. For each k, let $n' := \lfloor h(n)/k \rfloor$. We divide the interval $\{1, \ldots, h(n)\}$ into $k + 1$ subintervals as

$$I^{(n)}_{k,j} = \{(j-1) \cdot n' + 1, \ldots, j \cdot n'\} \ , \ j = 1, \ldots, k,$$

and

$$I^{(n)}_{k,k+1} = \{k \cdot n' + 1, \ldots, h(n)\} \ .$$

The subinterval $I_{k,k+1}^{(n)}$ may be empty if $h(n)$ is a multiple of k. When it is non-empty, the cardinality of this set is bounded by k, *i.e.,* $0 \leq |I_{k,k+1}^{(n)}| < k$. This is illustrated in the following figure.

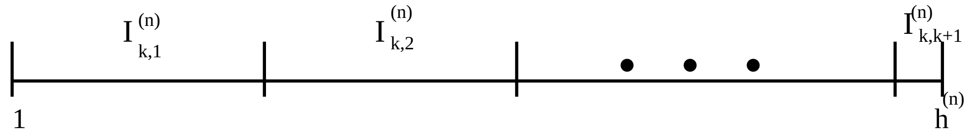


Figure 5.1: Construction of subintervals

The next mixing condition is the following:

Definition 5.2 *The array* $\mathbf{W}$ *is said to satisfy the condition* $D'(u_n)$ *if for all* $j = 1, 2, \ldots,$

$$\lim_{n\to\infty} \left(\sum_{i,i' \in I_{k,j}^{(n)} : i<i'} \mathbf{P}\left[W_i^{(n)} > u_n, W_{i'}^{(n)} > u_n\right] \right) = o\left(\frac{1}{k}\right) \quad , \tag{5.5}$$

for all $k \geq j$.

The interpretation of the condition $D'(u_n)$ is similar to condition given in (5.2). Since u_n is an increasing sequence in n, $\{W_i^{(n)} > u_n\}$ becomes a rare event as $n \to \infty$. Condition $D'(u_n)$ also states that the events $\{W_i^{(n)} > u_n\}$ do not happen in clusters with a high probability. The role of condition $D'(u_n)$ in the context of our problem will be discussed in the next section.

Notice that a sufficient condition for the condition $D'(u_n)$ to hold is

$$\lim_{n\to\infty} \left(\lfloor \frac{h(n)}{k} \rfloor^2 \cdot \sup_{i,i' \in I_{k,j}^{(n)} : i<i'} \mathbf{P}\left[W_i^{(n)} > u_n, W_{i'}^{(n)} > u_n\right] \right) = o\left(\frac{1}{k}\right) \quad , \tag{5.6}$$

for all $k \geq j$. The left hand side of (5.6) is an obvious upper bound of the left side of (5.5).

5.2.2 Distributional convergence

We now give the array $\mathbf{W} := \{W_\ell^{(n)}, n = 1, 2, \ldots; \ell = 1, \ldots, h(n)\}$ a new interpretation under the context of our problem. Let $h(n) = H(n)$, where $H(n)$ denotes the path hop count, and $W_\ell^{(n)}$ the inverse of the ℓ-th link's excess live, *i.e.*, $W_\ell^{(n)} = (X_\ell^{(n)})^{-1}$. With this relation between $X_\ell^{(n)}$ and $W_\ell^{(n)}$, it is obvious that to study the path duration $Z^{(n)}$, which is the minimum of $\{X_\ell^{(n)}, \ell = 1, \ldots, H(n)\}$, is equivalent to study the maximum of $\{W_\ell^{(n)}, \ell = 1, \ldots, H(n)\}$.

The Assumptions 4.1 and 4.2 are again imposed and two additional assumptions are needed.

Assumption 5.2 *For any sequence of intervals $\{\hat{I}^{(n)}, n = 1, 2, \ldots\}$,*

$$\frac{1}{H(n)} \sum_{\ell \in \hat{I}^{(n)}} \lambda_\ell^{(n)} = O\left(\frac{|\hat{I}^{(n)}|}{H(n)}\right) .$$

where $|\hat{I}^{(n)}|$ denotes the length of interval $\hat{I}^{(n)}$.

A *sufficient* condition for Assumption 5.2 to hold is that there exists some arbitrarily small positive constant δ such that the expected link durations satisfy $\mathbf{m}(G_\ell^{(n)}) \geq \delta$ for all $n = 1, 2, \ldots$ and $\ell = 1, \ldots, H(n)$. This condition holds if the expected link durations do *not* converge to zero as the network size increases. Since the link durations are more likely to depend on the nodes' mobility and their transmission ranges rather than on the network size, this is a reasonable assumption.

Lemma 5.1 *Under Assumption 4.2, for any fixed* $x > 0$*, the array* $\mathbf{W} = \{W_\ell^{(n)}, n = 1, 2, \ldots; \ell = 1, \ldots, H(n)\}$ *satisfies condition* $D(u_n)$ *with* $u_n = \frac{H(n)}{x}$ *for any* $x > 0$.

Proof: The proof is given in Appendix B.4. ∎

Assumption 5.3 *The array* $\mathbf{W} = \{W_\ell^{(n)}, n = 1, 2, \ldots; \ell = 1, \ldots, H(n)\}$ *satisfies the condition* $D'(u_n)$ *with* $u_n = \frac{H(n)}{x}$ *for any* $x > 0$.

With the relation between $W_\ell^{(n)}$ and $X_\ell^{(n)}$ introduced earlier in this section and the definition of the sequence $\{u_n = \frac{H(n)}{x}\}$, the event $\left\{W_\ell^{(n)} \geq u_n\right\}$ is then equivalent to the event $\left\{X_\ell^{(n)} \leq \frac{x}{H(n)}\right\}$. The condition $D\left(u_n = \frac{H(n)}{x}\right)$ given in for array $\{W_\ell^{(n)}, n = 1, 2, \ldots\}$ can be translated into an equivalent condition for array $\{X_\ell^{(n)}, n = 1, 2, \ldots\}$. In other words, condition $D(u_n)$ requires that as the hop distance between two sets of links along a path increases, the dependence in their excess lives measured by the difference given in (5.4) go away asymptotically.

Under any *loop-free* routing algorithm, we expect the geographic distance between two nodes to increase as the hop distance between them increases. Thus, conditions given in the form of increasing hop distance between two links can be interpreted as their increasing geographic distance. When two links become geographically far apart, it is reasonable to assume that dependence in their link excess lives becomes weak. The validation of condition $D(u_n)$ and $D'(u_n)$ using simulation under the RWP mobility models will be given in Section 7.1.

With these assumptions enforced, we can conclude the same distributional convergence result for the path duration.

Theorem 5.2 *Suppose that Assumptions 4.1, 4.2, 5.2 and 5.3 hold. Then, we have*

$$\lim_{n\to\infty} \mathbf{P}\left[H(n)\cdot Z^{(n)} \leq x\right] = \begin{cases} 1-e^{-\lambda x}\ , & \textit{if}\ \ x>0 \\ \\ 0\ , & \textit{if}\ \ x\leq 0 \end{cases} . \tag{5.7}$$

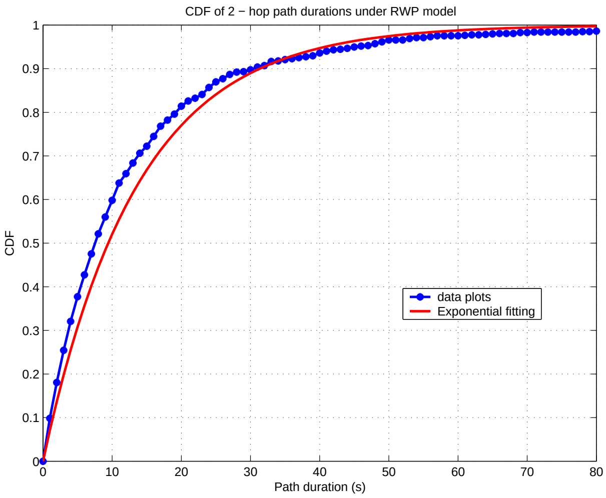
Proof: The proof is given in Appendix B.2. ■

Somewhat surprisingly, the parameter of the emerging exponential distribution in (5.7) is the same as that of the exponential distribution with independent link excess lives given in Theorem 4.1. Notice that under the mixing conditions, the dependence in link excess lives is not necessarily limited in a bounded neighborhood. This further suggests that when the path hop count is sufficiently large, the distribution of path duration is *not* significantly affected by the dependence in the reachability processes and link excess lives.

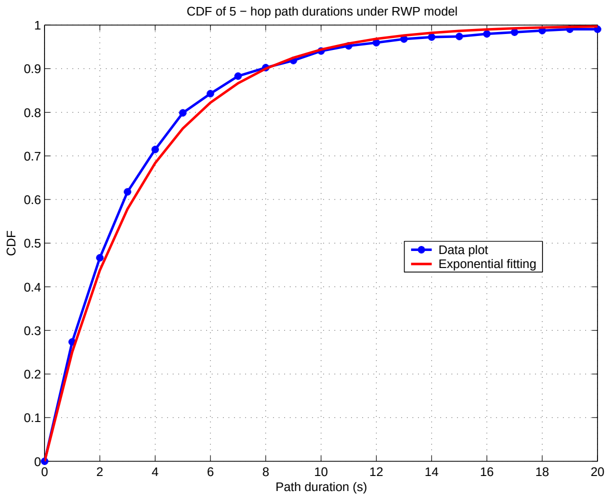
5.2.3 Simulation results

In this section, we present some simulation results that validate the convergence of path duration in distribution. The simulation set up is the same as described in Section 4.3.1.

Under the RWP mobility model, We plot the empirical CDFs of path distribution for the paths with 2 and 5 hops along with their exponential fitting curves (dash curves, obtained using the *expfit*($\cdot$) function in MATLAB statistics toolbox) in Figure 5.2 (a) and (b), respectively. The numbers of samples for 2 hop paths is 936 and 1342 for 5 hop paths. The fitting error calculated using the Kolmogorov-Smirnov test (K-S test)[1] decreases from 0.0823 for the 2 hop paths to 0.04952 for the 5 hop paths, which indicates better exponential fitting (convergence) for paths with larger hop count. Notice that the scales of the these two figures are different.



(a)

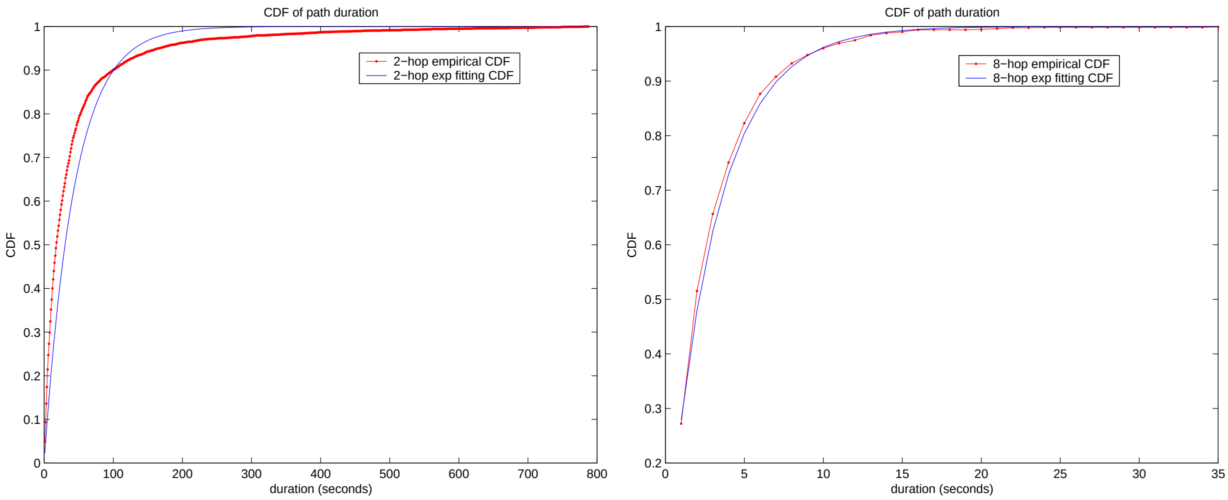


(b)

Figure 5.2: Exponential fitting of path duration distribution, (a) 2 hop path (b) 5 hop path

[1]Kolmogorov- Smirnov test error is taken to be $\sup_x |F_{\text{emp}}(x) - F_{\text{mle}}(x)|$, where F_{emp} is the empirical CDF of the data and F_{mle} is the exponential fitting function obtained via Maximum Likelihood Estimation.

The same trend in better exponential fitting of path durations for paths with increasing hop counts is observed from the simulation results under the RPG mobility model. This is shown in Figure 5.3 (a) for 2 hop paths and in Figure 5.3 (b) for 8 hop paths[2]. The K-S test fitting errors are 0.1272 and 0.0360, respectively.



(a) (b)

Figure 5.3: CDFs of path duration under RPG mobility model, (a) 2 hop path (b) 8 hop path.

In order to further justify that exponential distribution is a good fitting for the path duration distribution, we plot the natural logarithm of the empirical PDF of path durations for 2 hop and 5 hop paths under the RWP mobility model in Figure 5.4. As expected, both fitting curves for the PDFs have a negative linear slope. Although the measurements are noisy, we can still see that the data for 5 hop paths match the fitting curve better than the data for 2 hop paths.

[2] These figures are quoted from [46, Fig. 6.11] with the author's permission.

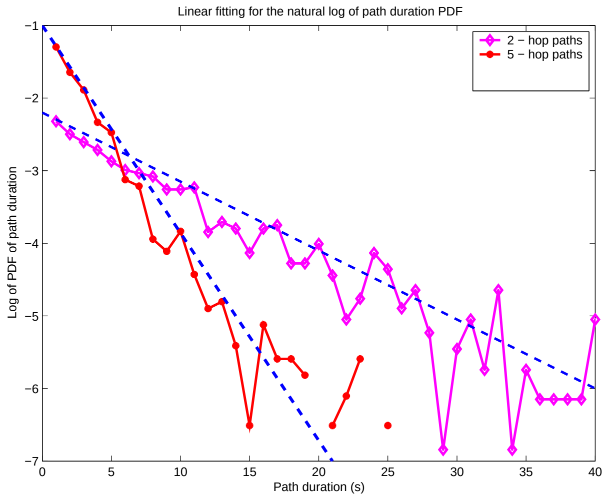


Figure 5.4: Logarithm of PDF of path duration under the RWP mobility model

Chapter 6

New path selection scheme – MED

Theorem 5.2 suggests that even with heterogeneous link durations, the distribution of path duration can be well approximated by an exponential random variable for sufficiently large path hop count. Our simulation results also show better exponential fitting for durations of paths with larger hop count under various mobility models. The next step is to exploit the relation between the link level statistics and the expected duration of path duration for the new path selection scheme.

The implication of our convergence result can be readily applied to real networks for the following reasons: First, it does not depend on the specific form of the distributions of link durations. In fact, in a large scale MANET, we do not expect the link durations to be homogenous, either. This is because that the network is likely to comprise many different types of nodes. These nodes may move around according to different mobility patterns, with different speed ranges. They may also have different communication capabilities, or simply different transmission ranges. Thus, the average link duration seen by individual nodes can vary widely from one node to another.

Second, our result claims that the approximation of the path durations by exponential rvs improves as the path hop count becomes larger. In a real network, as the network size increases, the hop distance between a randomly selected source-destination pair is likely to increase as well [14].

In short, for large scale MANETs, we can approximate the path duration by an exponential random variable for path with large hop count under a set of mild assumptions. Moreover, the parameter of the exponential rv can be used to estimate the expected path duration. Thus, it is possible to take into consideration the expected duration of a path along with its other qualities, *e.g.,* bandwidth, delay and etc., when we do path selection. In this dissertation, we mainly focus on the expected duration of path. We intend to select a path with the largest expected duration among other available paths[1]. With this objective, we propose a novel path selection scheme for large scale MANETs – *Maximum Expected Duration* (MED)

---

[1] In the current study, we assume that all candidate paths satisfy the requirements on other qualities of paths, *e.g.,* bandwidth, delay, etc.

path selection scheme.

One of the advantages of this path selection scheme is that its implementation into existing routing protocols requires little change to the current protocols. As we shall discuss in the rest of this chapter, its implementation into the AODV protocol only requires a few changes to the format of control packets and the way they are handled.

As an additional contribution of the MED path selection scheme, we are able to propose an efficient path recovery scheme that exploits the information obtained during the path discovery phase. This path recovery scheme is described in Section 6.2

6.1 Outline of MED

Theorem 5.2 implies that the inverse of the expected path duration, *i.e.,* the parameter of the emerging exponential distribution, can be approximated by the summation of the inverses of expected durations of the links along the path when the path hop count is large. In the rest of this section, we describe the two steps that lead to the path selection. The first step is the estimation of expected link durations. The second step is to allow the source collect necessary information used for estimating the expected path duration during the path discovery procedure.

6.1.1 Estimation of expected link durations

Estimation of the expected link durations is the first step towards estimating the path duration. A link $\ell = (i, j)$ is set up by two nodes, i and j. In order to obtain the expected duration of this link, say T, either node i or j needs to record the durations of this link whenever it is up, *e.g.,* $T_i, i = 1, 2, \ldots$ [2], and take an average of all T_i's. This approach is highly inefficient. The number of samples may not be large enough for a good estimation of T during the lifetime of the network.

Considering this difficulty, we use an alternative approach. Let each node in the network keep track of the *average* duration of all the links that it establishes with other nodes. When an estimation of the duration for a link is needed, one of the end node of the link uses its current value of the average it stores as an estimate. It is then obvious that an estimation for the expected duration of a link provided by its two end nodes will be different. How this discrepancy will affect the estimation of path duration is subject to further study.

This average link duration LD_{avg} stored at each node is updated whenever a link with one of its neighbors is terminated. There are different ways to update this average. For example, we can use exponentially weighted moving average. Let LD_{new} denote the duration of the link that just terminated, and LD_{avg}^{old} the average value before it is updated. Then, using exponential average, the updated average

[2] Here T_i denotes the duration of the link (i, j) for the i-th time it is up

LD_{avg}^{new} is given by

$$LD_{avg}^{new} = (1 - w)LD_{avg}^{old} + wLD_{new} \ ,$$

where w is some constant between 0 and 1. The parameter w for this updating rule determines the weight on the duration of the link that is just terminated.

It is clear that for each node, the durations of links it establishes with its neighbors depend on many factors, including its mobility (speed, direction), its location (edge or center of the network) and the mobility of its neighbors. Thus, each node may choose to classify its neighbors, based on their mobilities and/or transmission ranges, and maintain a separate average link duration for each type of neighbors.

6.1.2 Estimation of expected path duration

The second step in estimating the path duration is to provide the source with the estimates of expected durations of the links along the path. Since only the sum of these estimates is needed by the source, this information collection process can be added into existing on-demand routing protocols with little change to their path discovery process.

During a typical path discovery phase of an on-demand routing protocol, a source broadcasts its route request messages. If an intermediate node that receives this message knows about a path to the destination, then it will generate a path reply message. Usually, it will specify in the reply message information such as the hop count to the destination, the time to expire of the partial path, and etc. For our purpose, we add in addition the inverse of estimated expected duration of the sub-path to the destination in the reply message.

When implementing this scheme into the AODV protocol, a new field called *Inverse Path Duration* (IPD) is added to the path reply message. The node that generates a path reply message provides in the IPD field the inverse of the expected duration of the sub-path to the destination. Note that if it is the destination that generates the path reply message, then the initial value in the IPD field is set to zero.

After receiving a reply message, each upstream node updates its own expected duration of the sub-path to the destination if necessary. Its new estimation is given by the summation of the IPD value from the reply message and the inverse of its estimate of the expected duration of link on which this reply message is received. Then if needed, it forwards the path reply message towards the source with the updated estimation of the expected duration of the sub-path to the destination.

Finally, when the source node receives the path reply message, it also updates its estimate of the expected duration of the path. When multiple paths are discovered, it selects the one with the largest expected duration, *i.e.,* the one with the smallest IPD value, as the primary path for data transmission.

The detail of this algorithm will be given in Section 6.3 when we describe its implementation in AODV protocol.

6.2 Efficient local recovery scheme

When a link along a path in use fails, a local recovery procedure can be initiated by the node that detects the link failure. This is usually done through broadcasting route request messages again within a local neighborhood. If a node nearby happens to know a route to the destination that does not go through the failed link, then packet transmission can be resumed on this alternative path. In general, additional delays and energy consumption will be incurred during this path recovery process.

We now propose a local recovery scheme that avoids rebroadcasting path request messages if possible. When the MED scheme is used, there may be multiple paths to the destination cached at upstream nodes during the path discovery phase. When a link fails, the node that detects this link failure may already have an alternative path to the destination cached. If the probability this cached path is still available is high, then the node can start using this path to resume the data traffic without further checking its availability. Admittedly, if the cached path is indeed not valid when tried, then additional packet losses, delays and waste of energy can occur. Thus, accurate estimation of the probability the cached path is still available is important for the efficiency of this local recovery scheme.

To calculate the probability that a cached path is still available, we need to know the distribution of the path duration. Theorem 5.2 suggests that we can assume an exponential distribution for the duration of a cached path[3] and the parameter of this exponential distribution can be approximated by its IPD value. Suppose that the last time the path was known to be valid was t_0, which could be the time the entry was either created or updated. Then the probability the path is still available at time t_1, $t_1 > t_0$, is approximated by

$$\exp\{-\lambda(t_1 - t_0)\} \ , \tag{6.1}$$

where λ is the IPD value of the cached path.

The calculation of the probability that a cached path is still available using (6.1) only uses the a priori distribution of the duration of the cached path. However, since multiple paths cached at a node may actually share some common links, their path durations are not independent of one another. In particular, if the cached path being tried share some links with the primary path that just failed, then the probability that this cached path is still available is affected by the event that the primary path is already invalid. Here we simplify our problem by assuming that we can approximate the probability a cached path is available using (6.1). The dependency the durations of two paths that share some common nodes or links is subject to further study.

After calculating the probabilities that each cached paths are still available,

[3]For simplicity, we apply this exponential distribution approximation to all paths regardless of their hop counts, bearing in mind that path with larger hop count is better approximated by an exponential distribution.

we can do one of the following. If there are multiple cached backup paths with probabilities being available exceeding a chosen threshold value γ, then the one with the largest probability is chosen as the primary path[4]. If there exists no such path, *i.e.,* the probability any of the cached paths being available is below the threshold value γ, then the node either initiates a new path discovery procedure or sends an error message to upstream node(s). Once again, the goal of this scheme is to minimize the delay and the routing overhead incurred during local recovery procedures by choosing a backup path that is most likely to be available.

6.2.1 Updating the threshold γ

Under our local recovery scheme, each node attempts to use the cached path that may still be available with a high probability when the primary path fails. The goal of this approach is to reduce the delay and overhead caused by path failure. However, there are some issues we need to pay attention to. First, the probability that a cached path is available is computed according to (6.1) under the assumption that the path duration is exponentially distributed. Since our analytical result on the distributional convergence of path duration is an asymptotic one, for any path with finite hop count, there is a certain discrepancy between the assumed exponential distribution of path duration and the real path duration distribution. Second, the IPD value used for approximating the parameter of the exponential

[4] We assume that all these cached paths satisfy other requirements of path qualities

distribution is obtained through noisy measurements of the durations of the links along the path. Therefore, the real probability the cached path is available may be different from what is given by (6.1).

Although it is difficult to estimate the real probabilities that these cached paths are available, we expect that the probability a selected cached path is available can exceed some desired target probability, P_{target}, in the long run, if a proper threshold value γ is chosen.

Let $Q(\gamma)$ be an indicator function of the event that a selected cached path is available when the threshold is γ, $\gamma \in (0,1)$. The CDF of $Q(\gamma)$ is denoted by $H(\cdot)$, which is unknown. Let $m(\gamma)$ denote the mean of $Q(\gamma)$. Since as γ increases, it is reasonable that the portability a selected backup path is available increases, we assume that $m(\gamma)$ is a strictly increasing function of γ. Thus, our goal is to find a γ^* such that $m(\gamma^*) = P_{target}$, assuming its existence.

Assume that the current threshold value at node i is γ_n, $n = 1, 2, \ldots,$ which serves as an estimate of γ^* after trying n cached paths. A cached path at node i is tried at time t as an alternative path if probability of the path being available at time t is grater than γ_n. Then, node i updates its current threshold value γ_n based on the result of this trial. Let Q_n be an indicator function that $Q_n = 1$ if the backup path is indeed available, and 0 otherwise. The threshold updating rule for node i is the following:

$$\gamma_{n+1} = \gamma_n + \epsilon_n(P_{target} - Q_n),\ n = 1, 2, \ldots\ , \tag{6.2}$$

where ϵ_n is a positive step size. The initial value γ_0 is P_{target}.

It is known from stochastic approximation theory [6, 22] that under certain conditions on the sequence $\{\epsilon_n\}$, the sequence $\{\gamma_n\}$ converges to γ^* almost surely as n goes to infinity. Details of the proof are given in Appendix B.5.

6.3 Implementation of the MED scheme in the AODV protocol

In this section, we describe the implementation of the proposed path selection scheme into the AODV routing protocol. We begin with the description of the routing table structure. Then, we state in detail how the path request/reply messages are handled, as well as the route error message. At the end, we address the issue of loop avoidance.

Since we are not proposing a complete routing protocol, we do not attempt to minimize the overhead it generates. Such overhead is likely to depend on the routing protocol in which the proposed scheme is implemented. The additional overhead incurred by using MED scheme is mainly due to the discovery of multiple paths compared with original AODV protocol. Under the AODV protocol, nodes only respond to the first path request message they receive, and ignore any duplicate path request messages received afterwards. However, since our path selection scheme is based on the availability of multiple paths, we need to keep track of path request messages arriving from different upstream nodes.

Some features of the AODV protocol are kept as they were. For example, each node maintains an integer valued sequence number. It starts from zero, and is increased by one every time the node initiates a new path request message. When a node generates a path reply message, it includes its current sequence number in the message. Hence, the sequence number associated with each node contained in the route request and reply messages indicates the freshness of the routing information.

6.3.1 Routing table structure

For each known destination, there is a route entry that caches up to K paths. Each route entry is composed of K subentries. The destination ID is a common field in all three subentries. Each path information is stored in one of the subentries of the route entry.

Each subentry consists of seven fields: (i) Rank of the subentry, (ii) Destination sequence number, (iii) Next hop to the destination, (iv) Hop count to the destination, (v) Inverse Path Duration (IPD value of the path to the destination) (vi) path set up time, and (vii) expiration time of this route entry. The rank of the subentry determines whether this path is used as the primary path or not. This will be discussed shortly. We require that the next hop to the destination recorded in the K subentries to be different in order to avoid the redundancy of the path information cached.

The rank of each path, from 1 to K, indicates its preference. The path with the highest rank[5] is selected as the primary path. The rest of them may serve as backup paths when the primary path fails or for the purpose of multipath routing. Paths in a route entry are ranked first based on their destination sequence numbers, and then based on the IPD values. Ties are broken using the hop counts. In other words, the path information with a larger destination sequence number is always preferred. If two or more paths are for the same destination sequence number, then the one with a smaller IPD value takes a higher rank since it indicates a larger estimated *expected* path duration. Finally, if two paths still tie using their sequence numbers and IPD values, then the one with a smaller hop count is ranked higher. In Fig. 6.1, we illustrate the change of a route entry before and after its subentries are ranked.

Before ranking

Rank	Destination sequence #	IPD value	Hop count
1	52	0.5892	8
2	52	0.6578	9
3	50	0.3212	6
new path	54	0.5020	8

After ranking

	Destination sequence #	IPD value
1	54	0.5020
2	52	0.5892
3	52	0.6578
–	–	–

Figure 6.1: Ranking of discovered paths

[5]We take the rank 1 as the highest rank.

6.3.2 Handling path request messages

A path *request* message (RREQ) is broadcasted by the source when a path to a destination is needed and no such path is known. The RREQ message contains eight fields: (i) source ID, (ii) source sequence number, (iii) broadcast ID, (iv) destination ID, (v) destination sequence number, (vi) hop count to the source and (vii) time to live (TTL).

We explain some of the fields in the following. Each path request message is distinguished by its source ID and broadcast ID maintained by the source node. The broadcast ID is an integer and is increased by one each time the source generates a new path request message. The TTL value is an integer that indicates the number of hops this request message can be rebroadcasted. If there is no reply message received within certain amount of time, then the source rebroadcasts the request message after increasing the broadcast ID and the TTL value. For each destination, the TTL value in the request is initially set to be TTL_{ini}, and it is increased by a step size TTL_{δ} for every re-try of the path request. If no path is discovered after three attempts, the source node will wait a much longer time before next attempt. The hop count to source is initially set to one.

The source sequence number and hop count in the request message are used by intermediate nodes to update the reverse route to the *source*. When a node receives a RREQ message, it first updates the routing entry that contains the source node as the destination. It checks if there is a subentry for the source that uses the neighboring node from which the RREQ message is received as next hop. If so, it updates this route entry if one of the following is true: (i) The sequence number of the source in the RREQ message is higher than the sequence number in the subentry or (ii) the hop count in the RREQ message is smaller than the hop count in the subentry when the sequence numbers are equal. Otherwise, it creates a temporary subentry that sets the neighboring node from which the RREQ is received as the next hop. Finally, it ranks all the subentries along with the temporally subentry. If there are more than K of them, the subentry with the lowest rank is dropped.

In the following two cases, a node generates a path reply message in response to the RREQ message: (i) if the node is the requested destination or (ii) if the node has a route entry that contains a path to the destination with the destination sequence number greater than or equal to the destination sequence number requested in the RREQ message. Otherwise, the node shall perform one of the following. If this is the first time it receives this RREQ message (identified by its source ID and broadcast ID) and the TTL value in the RREQ message is greater than or equal to 1, then it shall rebroadcast the RREQ message with the hop count field increased by one and the TTL value decreased by one. If it has received another copy of the same request message before or the TTL value in the RREQ message is zero, then it terminates the RREQ message.

6.3.3 Handling path reply messages

As mentioned in Section 6.3.2, if an intermediate node knows about a path to the destination that has destination sequence number greater than or equal to the destination sequence number requested in the RREQ message, then it generates a path *reply* message (RREP). The RREP message contains four fields: (i) destination ID, (ii) destination sequence number, (iii) IPD value of the path to the destination, and (iv) the path hop count. The path information contained in the RREP message is that of the primary path to the destination known at the node. The intermediate node can send this RREP message along up to K_r[6] reverse paths to the source cached at the nodes. The RREQ message is then terminated.

When an RREQ message reaches the destination node, the destination node first raises its sequence number to the maximum of its current sequence number and the destination sequence number requested in the RREQ message. Then it generates an RREP message. The initial IPD in the RREP message is set to be zero and the hop count is one.

When an intermediate node receives a RREP message, it first calculates the updated IPD value of the path to the destination by summing the IPD value contained in the RREP message and its estimate of the expected duration of the link on which the RREP message is received. Then, it updates its routing information

[6]We let $0 < K_r \leq K$. In our simulation, we compared the performance of the path selection scheme with different values of K_r.

for the destination in the same way it updates the reverse path information back to the source. Please refer to Section 6.3.2 for the updating rules. The only difference here is that the IPD values of the paths in each subentry are now considered before the path hop count when these subentries are ranked. If the newly discovered path becomes the primary path to the destination, then the node generates an RREP message with the IPD value and hop count of the new primary path. Otherwise, the RREP message is terminated. Again, it can forward this RREP message to up to K_r upstream nodes cached at the node.

Finally, when the RREP message reaches the source, it updates the hop count and the IPD value of the new path and ranks it with other paths (if any exists). Data transmission can begin as soon as the first path to the destination is discovered. After processing a newly arrived RREP message, the source may switch its primary path to the newly discovered path that is considered to have longer expected path duration.

6.3.4 Handling path failures

When a link failure occurs, the node that detects the failure performs a local recovery as described in Section 6.2. It first checks if it has one or more cached paths that may still be available according to (6.1). If the probability that the cached path(s) being available exceeds certain threshold γ, then the one with the smallest IPD value is selected as the primary path immediately. A *route error* message (RRER) is broadcasted in either one of the following cases: (i) there is no cached backup path that satisfies the threshold condition or (ii) after trying a backup path, the node finds that it is actually not available.

The RRER message contains a list of IDs of the destination nodes that are no longer reachable due to the link failure. Associated with each destination node, there is a Local Recovery (LR) field indicating whether a cached backup path for this destination has been tried or not. The LR field is set to 1 if a cached path has been tried but was not successful; it is set to 0 if no cached path tried.

Each node that receives a RRER message does the following. For each destination listed in the RRER message, it first checks if there is any route subentry that uses the neighbor node from which the RRER message is received as the next hop. If so, this subentry is removed. If the removed subentry is not the one with the highest rank, then no additional action will be taken. However, if it is the primary path that is removed, then the node may need to perform a local recovery depending on the LR value for the destination. If the LR field is 1, which indicates that a local recovery has been performed and was not successful, then this node will simply rebroadcast this error information without doing anything. On the other hand, if the LR field is 0, which indicates that no cached path has been tried so far, then this node does a local recovery if it has any alternative path cached. Finally, the node broadcasts a RRER message containing the list of destinations that are still not reachable. Because of the use of LR field in the RRER message, the local recovery process is performed at most once for each destination affected by the link failure.

6.3.5 Loop avoidance

In this section, we describe the mechanism of preventing loops in the discovered paths. The information used for detecting loops is carried in the IP headers of the packets. In a practical implementation, the RREQ, RREP, and RRER messages mentioned in the previous sections are encapsulated in an IP packet. A typical IP header contains the ID of the node that sent this packet, the immediate next hop this packet needs to be forwarded to, and whether this is a broadcast packet or a unicast packet. If it is a broadcast packet, the immediate next hop ID field is void.

Under the AODV protocol, loops can be easily avoided. This is because each node only pays attention to the first RREQ message received. Any duplicated RREQ message received later from a different node or the same upstream node is discarded. However, our MED path selection scheme requires a node to cache multiple paths. As we will see shortly, this adds a couple of additional operation rules for handling these RREQ and RREP packets during the path discovery phase.

The loop can happen in three cases. The solution of preventing these types of loops will be given shortly.

1. The first type of loop happens during the process of disseminating path request messages. An example is shown in Figure 6.2. The source node

S broadcasts a RREQ message. Node i receives this RREQ message and records a reverse path to node S. It then broadcasts this RREQ message again. Node j after receiving this RREQ message, records a path back to source S through node i, and broadcasts the RREQ message. Node i will receive the RREQ message broadcasted by node j since it is one hop away from node j. Since it does not know that this RREQ message was trigged by its broadcasting, it will record another reverse path to source S through node j. A loop occurs.

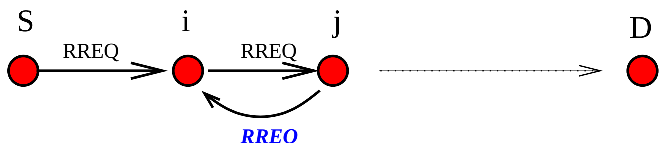


Figure 6.2: Loop of type one

2. The second type of loop also happens due to a similar reason for the type one loop, except that the loop is two hops instead of one hop. An example is shown in Figure 6.3. Again, the source node S broadcasts RREQ message. Node i rebroadcasts it. The loop in the reverse path that goes from node i to node k and back to node i (denoted by $i \rightarrow k \rightarrow i$) is of the first type as already discussed. Besides this loop, another loop in a reverse path from node i that goes though nodes $i \rightarrow k \rightarrow j \rightarrow i$ may also happen. These two types of loops are the main sources of loops during the process of broadcasting path request messages. There could be similar types of loops but of larger hop

count. However, since the reverse paths with loops always have a larger hop count then that of a loop-free one, they will not be selected as the primary path.

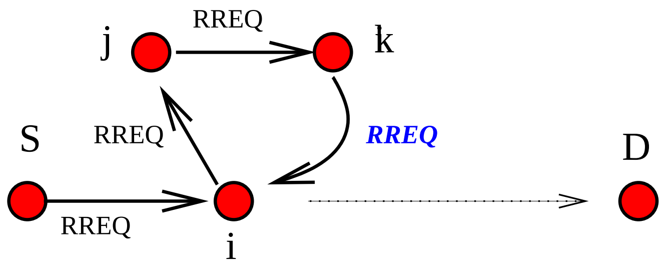


Figure 6.3: Loop of type two

3. The third type of loop occurs during the process of forwarding RREP messages back to the source. An example is shown in Figure 6.4. Suppose that node j receives a RREP message from the destination D. It forwards this RREP message to two of its upstream nodes i and k back towards the source. The RREP message forwarded to node k will not cause any problem since node k has only one upstream node. However, the RREP message forwarded to node i may trigger a loop. If node i has received RREQ messages from both node m and node j earlier, then it considers node j as an upstream node leads to the source. Thus, when it tries to forward a RREP message, it forwards the RREP message to node j, even though this RREP message may actually be received from node j. A loop then occurs.

   Although the a path with loop in this case will certainly have a larger IPD value and a larger hop count, thus will not be chosen as the primary path,

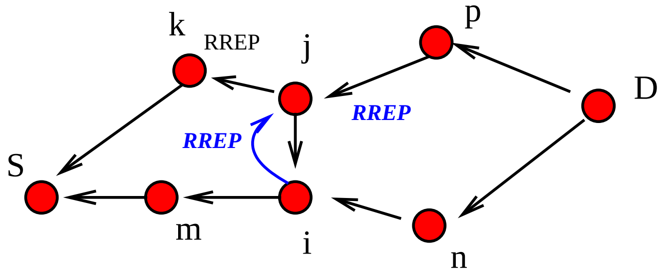


Figure 6.4: Loop of type three

it may still be cached as a backup path. In the example shown in Figure6.4, node j after receiving the RREP message forwarded from node i may cache node i as another next hop leading to the destination. However, this path node j caches indeed goes through node j again. Thus, when the primary path (assumed to be $j \to p \to D$ fails due to the link failure between node j and p, node j will try to use the backup path $j \to i \to j \to p \to D$, which involves a loop and is not available either.

The solutions to detecting and preventing these three types of loops require additional information related to the nodes' identity along the path to be known to the intermediate nodes.

1. Solution to type one loop:

   This type of loop is caused by the fact that node i does not know that the RREQ message it receives from node j is trigged by itself. Hence, when a node rebroadcasts a RREQ message, in addition to the ID of itself that is contained in the IP header of the packet, it also put the ID of the node from which the RREQ message is received. Thus, when node i receives the RREQ message from node j, it can check the IP header of the packet and avoid this one hop loop.

2. Solution to type two loops:

   The solution to the type one loop also solves the problem that causes type two loops to some extent, as illustrated in Figure 6.5. The type two loop goes through nodes $i \rightarrow k \rightarrow j \rightarrow i$ (path 1). Since node k is only one hop away from node i along path 2 and two hops from node i along path 1, we expect that the RREQ from node i along path 1 reaches node k later than the RREQ from node i along path 2. If so, since node k only rebroadcasts the first RREQ message[7] it receives, *i.e.,* the one from node i, but not the ones received later on, it will not rebroadcast the RREQ message received from node j. Thus, node i will not be aware of a loop that goes through path 1. Therefore, preventing loop of type one also helps to exclude loops of type two.

3. Solution to type three loop:

   The solution to the third type is also simple. Before a node forwards a RREP message, it first checks if the next hop to the source is also the node from which the RREP message is received. In the example shown in Figure 6.6,

[7] Recall that each RREQ message is identified by the source ID and broadcast ID in the message.

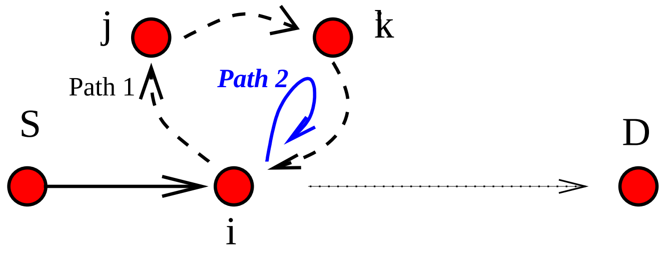


Figure 6.5: Solution to loop of type two

node i both receives and forwards RREP message from and to node j. If so, this node should simply not forward the RREP message. Notice that this does not imply that node i shall not forward any RREP message to node j in our example. There could be another path from node D to node i that does not go through node j. In this case, the RREP from node n should be forwarded to node j by node i.

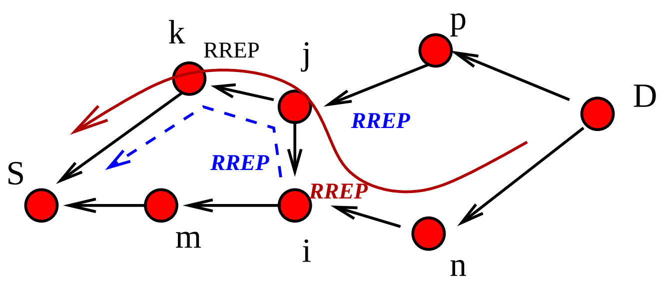


Figure 6.6: Solution to loop of type three

Chapter 7

Simulation results

In Chapters 4 through 6, we derived some analytical results on the path duration in large scale MANETs. Based on these analytical results we proposed a new path selection scheme. In this chapter we use some simulation results to further justify our study. We start with validating some assumptions imposed in Chapter 5 in Section 7.1. Then we evaluate the performance of our integrated protocol (the MED scheme implemented in the AODV protocol) and compare it with that of the AODV protocol in Section 7.2. We discuss the impact of the value of K_r, the number of paths to forward the RREP message in the MED scheme, in Section 7.3.

The simulator used for this study is called NS-2. NS is a discrete event simulator that provides substantial support for simulation of TCP/IP, MAC and physical layer protocols over wired and wireless networks. NS-2 is the second version of it.

For information about NS-2, please refer to [51]. We need to point out that information related with the mobility of nodes, *e.g.,* the location of nodes on the network domain at all times during the simulation time are recorded by a General Operations Director (GOD) object. Generally speaking, NS-2 uses this GOD object to store global information about the state of the network that an omniscient observer would have. These global information should not be made known to any participant (or node) in the simulation.

Some common settings of our simulations are summarized here: The simulations are run on a rectangular region of 2 km $\times$ 2 km. There are 200 nodes moving across this region. The transmission range of all the nodes are fixed as 250 m. Hence, the link between two nodes is considered up if the distance between these two nodes is less than 250 meters. Each node moves across this region according to the RWP mobility model. Each simulation run lasts for 1,200 seconds. In order to reduce the effects of transient period of network performance, data are collected only during the last 800 seconds of each run.

7.1 Validation of assumptions

In this section, we validate the assumptions on the dependency of link excess lives imposed in Chapter 5. For the finite dependence case, we validate condition (5.2). This condition requires that given one link having a very small excess life, the probability that any of its neighboring link within m hop distance has an excess life no larger than that of this one is very small. For general dependence case, we validate conditions $D'(u_n)$ for the set of excess lives of the links along a path.

7.1.1 Validation of condition (5.2)

In addition to the set up described in the previous section, the speed range of nodes is $[1, 10]$m/s.

To verify condition (5.2), it is sufficient to check if for any integer value $k > 0$, the conditional probability $\mathbf{P}[X_{\ell+k} < c | X_\ell < c]$ goes to zero as $c \to 0$, where X_ℓ and $X_{\ell+k}$ denote a pair of link excess lives of links with hop distance k. We plot in Figure 7.1 this conditional probability for $k = 1, 2, 4$. The $x-$axis is the value of c. It is obvious from the figure that these conditional probabilities lie on top of each other and all go to zero as c decreases to zero. Although we do not show here all the figures for all values of k we examined, the same observation is made for those cases. Thus, at least under the RWP mobility model, the condition (5.2) is easy to satisfy.

7.1.2 Validation of condition $D'(u_n)$

The condition $D'(u_n)$ is introduced in Section 5.2.1. The role and interpretation of this condition are similar to those of condition (5.2) in the finite dependence case.

The sufficient condition of this condition $D'(u_n = \frac{H(n)}{x})$ given in 5.6 can be

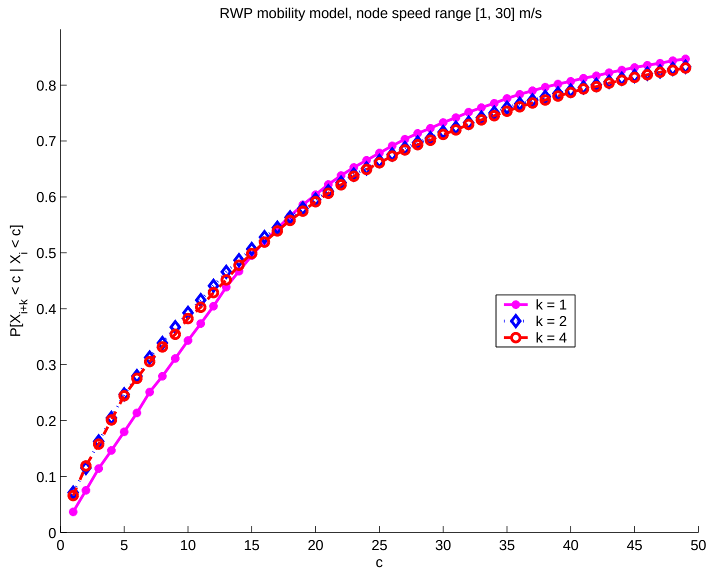


Figure 7.1: Plot of the conditional probabilities $\mathbf{P}\left[X_{\ell+k} < c | X_\ell < c\right]$, $k = 1, 2, 4$

written as the following (with $h(n)$ replaced by $H(n)$ and using the relation $W_\ell^{(n)} = (X_\ell^{(n)})^{-1}$):

$$\lim_{n\to\infty}\left(\lfloor\frac{H(n)}{k}\rfloor^2 \cdot \sup_{i,i'\in I_{k,j}^{(n)}:i<i'} \mathbf{P}\left[X_i^{(n)} < \frac{x}{H(n)}\right]\mathbf{P}\left[X_{i'}^{(n)} < \frac{x}{H(n)} | X_{i'}^{(n)} < \frac{x}{H(n)}\right]\right)$$
$$= o\left(\frac{1}{k}\right) \qquad \text{for all } k \geq j \ . \tag{7.1}$$

As mentioned in Section 5.2.2, a sufficient condition for Assumption 5.2 to hold is that there exists some arbitrary positive δ such that $\mathbf{m}(G_\ell^{(n)}) \geq \delta$ for all $n = 1, 2, \ldots$ and $\ell = 1, \ldots, H(n)$. Combine this condition with the relation given in (3.9), we have

$$\mathbf{P}\left[X_i^{(n)} < \frac{x}{H(n)}\right] = O(\frac{x}{H(n)}) \ .$$

If the conditional probability $\mathbf{P}\left[X_{i'}^{(n)} < \frac{x}{H(n)} | X_{i'}^{(n)} < \frac{x}{H(n)}\right]$ is also on the order of $O(\frac{x}{H(n)})$, (*e.g.*, when link excess lives are mutually independent,) then (7.1) holds.

In the following, we validate the condition that

$$\mathbf{P}\left[X_{i'}^{(n)} < \frac{x}{H(n)} | X_{i'}^{(n)} < \frac{x}{H(n)}\right] = O(\frac{x}{H(n)}) \tag{7.2}$$

by showing

$$\mathbf{P}\left[X_{i'}^{(n)} < c | X_{i'}^{(n)} < c\right] < \mathbf{P}\left[X_{i'}^{(n)} < c\right]$$

for small value of positive c, since $\frac{x}{H(n)} \to 0$ as $n \to \infty$.

The conditional probabilities $\mathbf{P}\left[X_{\ell}^{(n)} < c | X_{\ell+k}^{(n)} < c\right]$ for $k = 2, 4$ have been plotted in Figure 7.2. They are compared with the plot of probability $\mathbf{P}\left[X_{\ell}^{(n)} < c\right]$. The $x-$axis is the value of c. We can see from the figure that for smaller values of c, these conditional probabilities always lie under the unconditional probability, validating (7.2) and hence the condition $D'(u_n)$.

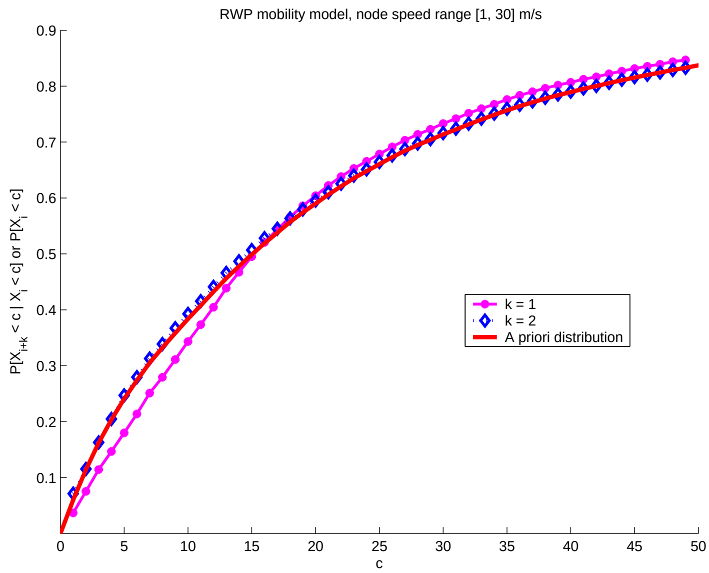


Figure 7.2: Validation of condition $D'(u_n)$

7.2 Performance evaluation of MED

In this section, we present simulation results on the durations of the primary paths selected using the MED scheme and other criteria, *e.g.,* min-hop path selection[1]. The simulations are run on a MANET with different types of nodes, moving in different speed ranges. The motivation for studying such a network is twofold. First, a real network is likely to have different types of nodes, *e.g.,* A Future Combat System will include different types of vehicles (e.g., jeeps, tanks, etc.), soldiers, and possibly aerial vehicles. These nodes may have different speed range and mobility patterns. Second, we mentioned in Chapter 6 that when multiple types of nodes exist in a network, a node can choose to maintain separate average link durations for different types of neighbors or simply combine them into one average. We are interested in finding out how much improvement can be achieved when neighbors are classified.

We generate two types of nodes, both moving according to the RWP model but with different speed ranges. Nodes of type one move at a speed chosen uniformly over the speed range $[1, 5]m/s$. Nodes of type 2 have a speed range of $[10, 30]m/s$.

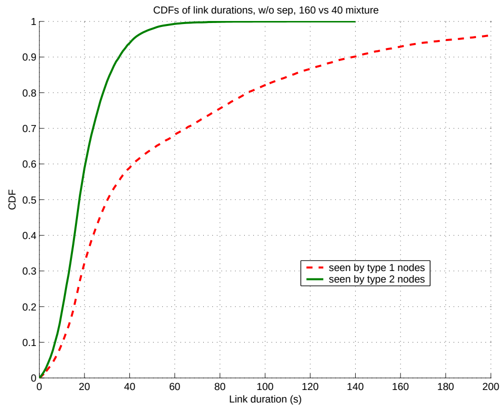
We run the simulation under three scenarios, with different combinations of the numbers of type 1 and type 2 nodes. The total number of nodes is fixed at 200.

[1] Min-hop path selection scheme refers to a path selection scheme that selects the path with the minimum hop count among available ones

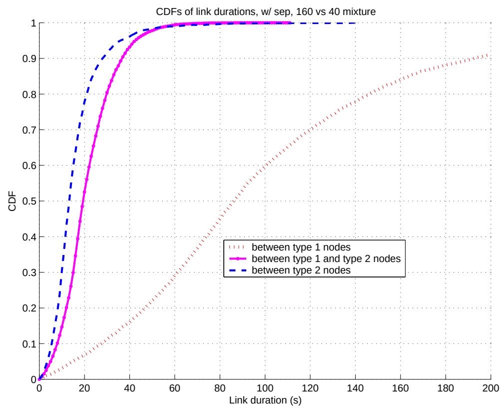
We begin with 140 type 1 nodes, and then increase it to 160 and finally to 180 [2]. For each scenario, we run the simulation under two different modes. In the first mode, each node maintains a single average link duration for all neighbor nodes. This is shown as "w/o sep" mode in the caption of the figures in the rest of this section. In the second mode, each node classifies the neighbors and maintains two separate averages, which is shown as "w/ sep" later on.

Since we fix the transmission range of all nodes (regardless of their type), slower moving type 1 nodes in general experience longer link durations than faster moving type 2 nodes. This can be seen in Figure 7.3. In Figure 7.3 (a), we plot the CDFs of link durations under the MED path selection scheme for the 160 vs. 40 node mixture scenario in the "w/o sep" mode. We plot the CDFs of the durations of links recorded by type 1 nodes and type 2 nodes. As expected, the link durations experienced by type 1 nodes are much larger (in stochastic order) than those experienced by type 2 nodes. In the "w/ sep" mode, we plot in Figure 7.3 (b) the CDFs of the link durations between (i) two type 1 nodes, (ii) a type 1 node and a type 2 node, and (iii) two type 2 nodes in three dash curves. It is interesting to note that the difference in the CDFs of link duration distribution of (a) between two type 2 nodes and (b) between a type 1 node and a type 2 node is not very large.

[2] The number of type 2 nodes are thus equal to 60, 40 and 20 in the three scenarios, respectively



(a)



(b)

Figure 7.3: CDFs of link durations

7.2.1 Improvement in primary path durations

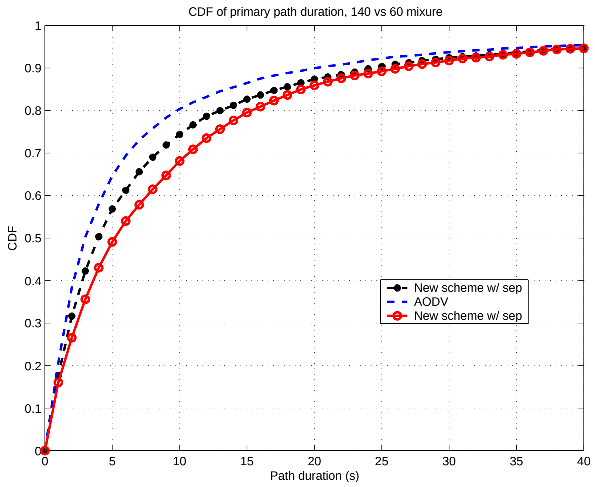
For each scenario, a total number of 10,000 connections are requested between randomly selected pairs of source and destination nodes. Every connection request generates a path request message that triggers a path discovery phase. For each path discovered, its set-up time and IPD value are recorded. The inter-arrival time of two connection requests are given by independent exponential random variables with rate 1. In other words, in each time slot (of duration 1 second), the number of path requests is distributed according to Poisson distribution with rate 1. Since we only generate path request messages within the last 800 seconds of the simulation time, we have about 800 paths set up for each run. We repeat the simulation 13 runs until 10,000 path requests are made.

During the simulation time of each run, all nodes continuously monitor the status of the links with all their neighbors by broadcasting "Hello" messages three times a second and listening to the Hello messages from its neighbors. At any time, each node maintains the inverse of average durations of all its links established in the past. We call this Inverse of the Average of the Link Durations (IALD). Whenever a node is involved in forwarding path reply messages, it updates the IPD value in the reply messages by simply adding its current IALD value to it.

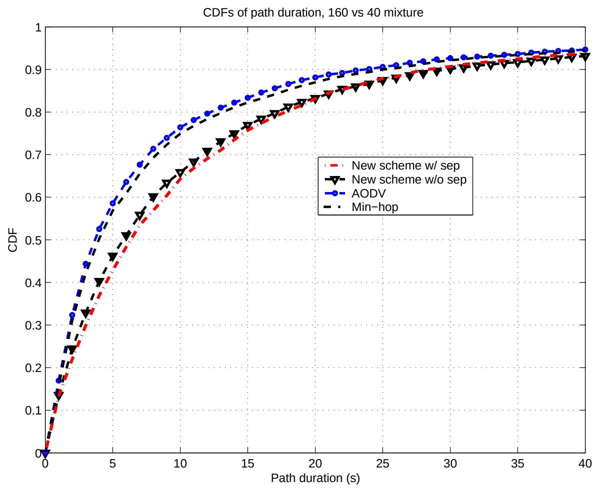
The CDFs of the primary path durations under the integrated protocol and the AODV protocol are plotted in Figure 7.4. Figure 7.4 (a) corresponds to the 140 type 1 nodes vs 60 type 2 nodes scenario. From left to right, we plot CDFs of path duration obtained (i) under the AODV protocol in the dash blue curve, (ii) under the MED "w/o sep" mode in the solid black curve, and (iii) under the MED "w/ sep" mode in the dash line with a star. It is obvious that the CDFs of path durations under our new scheme lie to the right of that under the AODV protocol. This means the durations of primary paths under our scheme are stochastically larger than those under the AODV protocol. The same observation can be made in Figure 7.4 (b) and (c), which are for the 160 type 1 nodes vs. 40 type 2 nodes and 180 vs. 20 scenarios.

The improvement on path duration under our integrated protocol is close to what we expected. Under the AODV protocol, nodes only focus on the first path request message received, thus the selection of path depends heavily on the arrival order of routing messages. Thus, this scheme cannot identify the mobility of nodes. As seen in Figure 7.3, slower moving nodes in general experience longer link duration (hence longer link excess live), we should benefit from establishing paths using mostly slower moving nodes. Under our integrated protocol, since the IALD value characterizes the average link duration a node experiences, a node with smaller IALD value is more likely to be of type 1. Therefore, when the percentage of type 2 nodes is large, *e.g.,* 60 out of 200, it is quite beneficial to improve the path duration by selecting the path with fewer links involving type 2 nodes.

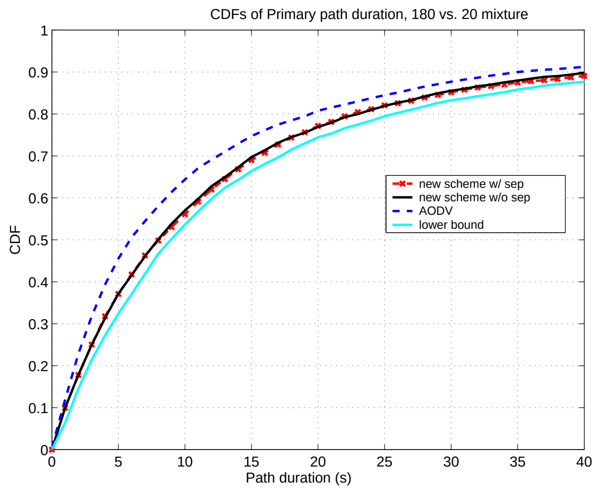
From the plots in Figure 7.3, we observe a trend that as the number of type 2 nodes is reduced to 40 and then to 20, the benefit of the integrated protocol compared with the AODV protocol decreases. This is simply because as type 2 nodes become more sparse in the network, a path is less likely to involve type 2 node naturally. For the same reason, maintaining separate averages at the nodes helps better to avoid the link(s) between two type 2 nodes when we do path selection, and hence provide further benefit. In the 180 vs 20 mixture scenario, in which only 10% nodes are of type 2, most links are set between type 1 nodes. Thus, the benefit under the “w/ sep” mode diminishes.



(a)



(b)



(c)

Figure 7.4: CDFs of path duration

We also compare the MED path selection scheme with min-hop path selection. For the 160 vs. 40 nodes mixture scenario, we run the simulation by changing the IALD value to be a constant for all nodes at all times. Thus, the path with minimum IPD value is also the one with minimum hop count. There may be more than one path with the same hop count, but they are viewed as equivalent under the min-hop criterion. The CDF of durations of primary path selected based on min-hop criterion plotted in a dotted dash line (magenta) in Figure 7.4(b). Again, the path durations are larger under our scheme in the stochastic order than those under min-hop criterion. It is interesting to note here that this improvement in path duration is not achieved by increasing the hop count of the primary paths. As shown in Figure 7.5, although the hop counts of the paths selected by the min-hop criterion are smaller than those of our scheme and the AODV protocol in the stochastic order, the differences between them are in fact insignificant. This means that the MED algorithm outperforms other path selection criteria by successfully identifying and avoiding links with potentially short excess lives, but not by going through paths with unnecessarily longer hops.

In order to evaluate the performance of our path selection scheme with some bench mark, we plotted the CDF of durations of paths selected by min-hop criterion in a MANET with only type 1 nodes. This is plotted in Figure 7.4 (c) in comparison with the 180 vs. 20 nodes mixture scenario. Since nodes are homogeneous, the min-hop path selection criterion is equivalent to our algorithm as explained in the

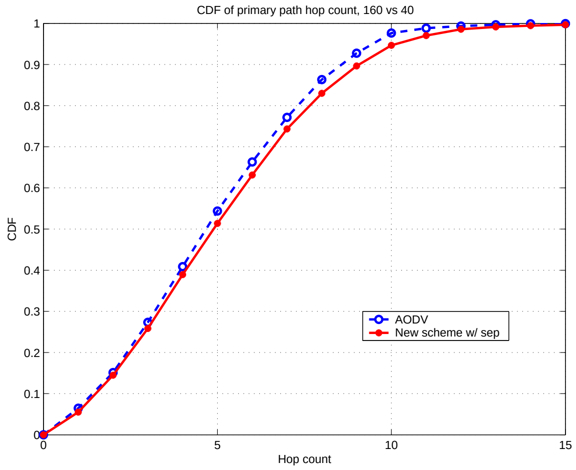


Figure 7.5: CDFs of hop counts for primary paths

previous paragraph. Thus, the CDF of path duration under this scenario serves as a "lower bound". As shown in the plot, the CDF of path duration under our scheme is very close to the lower bound, which indicates the efficiency of our scheme.

7.2.2 Quantitative comparison

To highlight the quantitative improvement in terms of path durations shown in Figure 7.4, we summarize some of the data points read from the figure. First, the median values of path durations are given in Table 7.1. A median value x for a rv X is such that $\mathbf{P}[X \leq x] = 0.5$. In all scenarios, we see an obvious increase in the media values. In particular, for the first two scenarios, the increase in the median values of the path duration under the "w/ sep" mode using the MED scheme over the AODV protocol is close to 62 percent. In the third scenario, the increase is near 40 percent. The improvement in the median values from the "w/o sep" mode to the "w/ sep" mode decreases from the first scenario to the third scenario. The reason for this decrease has been explained in the previous section.

Table 7.1: Median values of path durations

type 1 & 2 nodes	AODV	w/o sep. avg.	w/ sep. avg.
140 & 60	2.98	3.97	4.96
160 & 40	3.68	5.80	5.98
180 & 20	5.90	8.00	8.10

Another aspect of these plots we are interested in is about the left lower area of the CDF curves. This is where we have very small path durations, hence incur frequent path failures. To improve the network serve performance, it is of much importance to minimize the number of short-lived paths. We examine the probability $\mathbf{P}\left[\text{path duration } \leq x\right]$ for some small values of x. For example, consider the case $x = 3$ second. The probabilities $\mathbf{P}\left[\text{Path duration} \leq 3 \text{ seconds}\right]$ for the three scenarios are given in Table 7.2. It reads from the table that this probability is reduced by 45%, 51% and 22% under the "w/ sep" mode of our scheme than under the AODV protocol.

Table 7.2: Probability of path duration less than 3 seconds

class 1 & 2 nodes	AODV	w/o sep. avg.	w/ sep. avg.
140 & 60	0.5024	0.4224	0.3784
160 & 40	0.4436	0.3288	0.3166
180 & 20	0.3179	0.2496	0.2496

7.3 Impact of parameters K and K_r on path selection

In the previous section, we showed significant improvement in primary path durations using the new path selection scheme. One of the reasons that made this improvement possible is that multiple paths are being discovered, cached and compared. In our MED scheme, we allow each node cache up to K paths to a destination node. These paths all have different next hop nodes. This means that each intermediate node can cache up to K reverse paths to the source. When an intermediate node needs to forward a RREP message to the source, it can choose K_r out of K paths along which to send the RREP message. It is interesting to find out how the choice of these parameters affects the performance of the path selection scheme.

Intuitively, the number of paths can be discovered by the source node increase with K_r for a fixed K. We run the simulation under the 140 vs. 60 and 160 vs.

40 nodes mixture scenarios as described in Section 7.2 with $K_r = 1, 2$ and 3[3]. The average number of paths discovered by the source is very close to 1 when $K_r = 1$. It increases to about 3 when $K_r = 2$. However, when $K_r = 3$, the average number of paths discovered by the source is increased to 9.

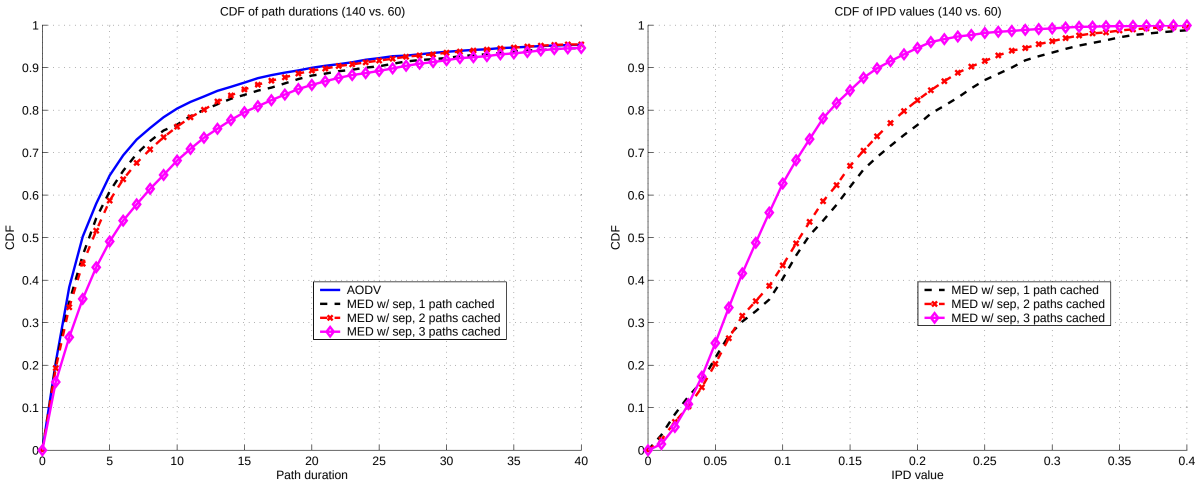
Moreover, when we compare the distribution of primary path selected for different values of K_r's, we find that the average duration of primary paths selected when $K_r = 3$ is much larger than that of primary paths selected when $K_r = 1$ or 2. In Figure 7.6 (a) and Figure 7.7 (a), we plot the CDFs of the primary paths using the AODV protocol and our new path selection scheme[4] when $K_r = 1, 2$ and 3, respectively. The plot in Figure 7.6 corresponds to the scenario with 140 vs. 60 node mixture, and the plot in Figure 7.7 is for the 160 vs. 40 mixture scenario. In both cases, we observe a significant improvement in the path durations under our new protocol from $K_r = 1$ or 2 to $K_r = 3$. In Figure 7.6, there is almost no difference between the plots for $K_r = 1$ and 2. This is consistent with the observation made on the IPD values of these primary paths selected. In Figure 7.6 (b) and Figure 7.7 (b), we plot the CDFs of the IPD values of the paths. We see from the figures that when $K_r = 3$, the IPD values of the primary paths tends to be much smaller than those of primary paths when $K_r = 1$ or 2.

We also see from the plots that even when only one path is cached, *i.e.*, $K_r = 1$,

---

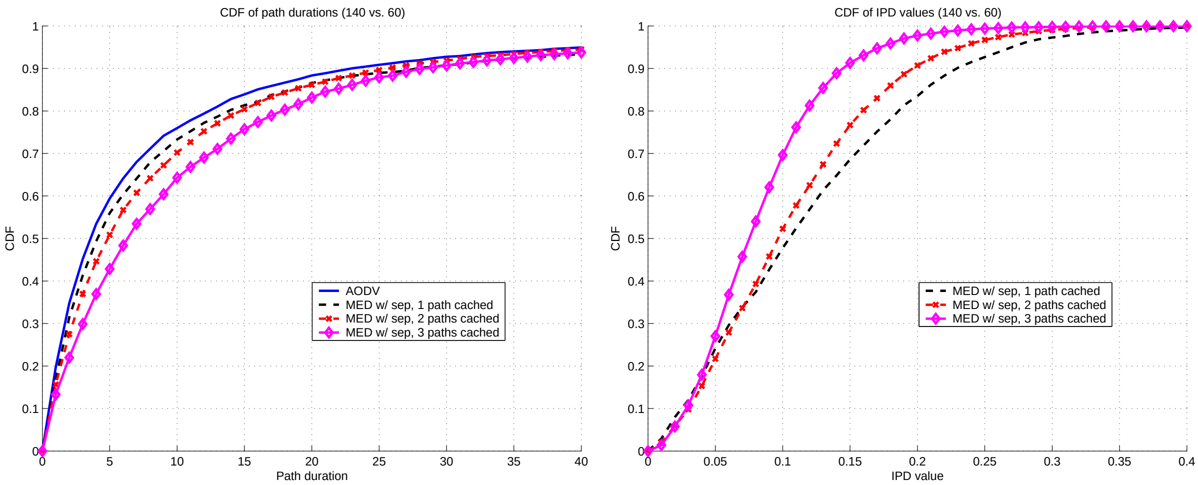
[3]In the simulation, the value of K is set to be equal to K_r. This is because we choose the K_r paths with rank from 1 to K_r for forwarding the RREP messages.

[4]We run the simulation in the "w/ sep" mode described in Section 7.2.



(a) (b)

Figure 7.6: Scenario 1: 140 vs 60 mixture, $K_r = 1, 2, 3$



(a) (b)

Figure 7.7: Scenario 2: 160 vs 40 mixture, $K_r = 1, 2, 3$

our path selection scheme still outperforms the AODV protocol. This suggests that although there are only a few times that multiple paths are discovered, when we make use of their IPD values we can still improve the average duration of primary paths.

These observations from the figures suggest that when only 1 or 2 paths are used for forwarding RREP messages, the chance for a path with the smaller IPD value to be discovered is not large. In order to understand this issue, we must first look what happens when we only cache one path. First, when we cache reverse paths to the source, only the source sequence and the hop count information are used to determine the preference of paths[5]. A RREQ message (with the same sequence number) received later on from a different upstream node will not trigger an update on the reverse path information if the hop count of the path is the same as what is currently stored. Thus, when there is only reverse path chosen as the path along which the RREP message will be forwarded, the node will use the path discovered earlier and with a smaller hop count. Since the hop count is not related to the IPD value of a path in any obvious way, the path with the minimum hop may not be preferable.

Secondly, even if we include the IPD value of the partial path to the source into the RREQ message and let each intermediate node rank paths based on their

[5]The IPD value of the partial path towards the source is not currently contained in the RREQ messages

IPD values, we may still miss the path with the smallest IPD value. This is because a RREQ messages received later on with a smaller IPD value will not be rebroadcasted if a RREQ message with the same source node ID and broadcast ID has been rebroadcasted. Thus, the intermediate nodes on the downstream side of the path may not learn about this "better" path, and hence they may not forward a reply message back to the source along this "better" path. This is illustrated in the following figure.

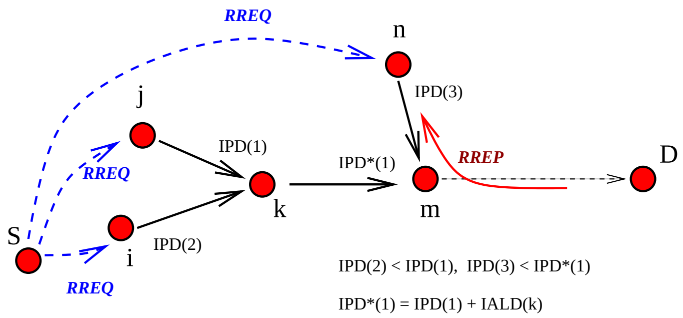


Figure 7.8: Disadvantage when $K_r = 1$

In this example, assume that the path with the smallest IPD value is $S->i->k->m->D$. During the path discovery phase, both node i and node j rebroadcast a RREQ messages that are received from node S. Assume that the one from node j arrives at node k earlier than the one from node i. Node k after receiving the RREQ message from node j, immediately rebroadcasts it to node m. However, it will not inform node m about the later discovered reverse path through node i after it receives the RREQ message from node i. Node m also receives a RREQ message from node n besides the RREQ message from node k.

Suppose that the IPD value learned from node k, IPD*(1) [6] in Figure 7.8, is larger than the one from node n, IPD(3) in the figure. Then, when a path reply message arrives from the destination, node m will forward it to node n instead of node k. Hence, the path with the smallest IPD will not be discovered by the source.

[6] This IPD value is the sum of IPD(1) and IALD(k), where IALD(k) is the inverse of estimated duration of link (k, n) maintained by node k.

Chapter 8

Conclusions

In this dissertation, we have presented theoretical results on the asymptotics of path durations for large scale MANETs. Some simulation results are presented to verify the analytical results. A new path selection scheme is proposed based on these findings. We now summarize our results and discuss their implications in real networks.

8.1 Analytical results

First, our main result is that for a large scale network, if the dependency between link excess lives decreases as the hop distance between them increases, then the path duration, properly scaled, converges to an exponential random variable. Moreover, the parameter of the emerging exponential random variable, which can be used to approximate the expected path duration, is related with the expected durations of the links along the path through the sum of their inverses[1].

Second, we propose a new path selection scheme that selects paths based on their expected durations. This path selection scheme exploits the relation between path duration and link level statistics just mentioned. Moreover, it can be easily implemented into existing routing protocols with little change to the handling of routing control packets. The information needed for the source node to estimate the expected duration of a path can be collected during the path discovery phase.

We need to point out that this strategy relies on an implicit condition that nodes are capable of monitoring the durations of the links with its neighboring nodes. In the dissertation, we proposed to let each node constantly monitor the "Hello" messages sent from its neighboring nodes. It can approximate the durations of the links established with its neighbors using the number of "Hello" messages recorded, assuming the intervals between two "Hello" messages are known.

8.2 Simulation results

We now summarize the simulation results. First, the assumptions we introduced to prove the convergence result on path duration are validated under the RWP mobility model. In the general dependent case discussed in Section 5.2, we

[1]This is different from what was assumed by the ABR routing protocol [43] that the path duration is longer if the average age of the links is larger. The age of a link is defined as the amount of time the link is on when a path is set up.

validated the two mixing conditions on the set of excess lives of links along a path. These assumptions have also been validated under other mobility models, *e.g.,* FH, MA, RPGM. Please refer to [46] for more details. Therefore, these assumptions appear reasonable for large scale MANETs.

Secondly, for paths with large hop count, the path durations can be well approximated by an exponential random variable. The accuracy of the approximation improves as the path hop count increases. This is consistent with the findings in [40, 46].

Thirdly, the improvement in the durations of primary paths selected by the MED path selection scheme can be affected by the parameters used in its implementation. In our simulations, we compare the CDFs of primary path durations when the number of paths along which an intermediate node can forward RREP messages varies from $K_r = 1$ to 3. As the value of K_r increases, our scheme shows more improvement in the durations of primary paths compared with those under the AODV protocol.

8.3 Future work direction and comments

There are some interesting directions for future work. First, we can develop a more analytical model on the link durations, especially on their dependency relations. This dependency is likely to depend on the mobility models used. Some analytical study on mobility models can be found in [2, 4, 15, 31].

Second, in our implementation of MED in the simulation, the average of link duration each node maintains is taken by an arithmetic mean. As proposed in Section 6.1.1, we may update this average by using an exponential averaging, which poses different weight on the duration of the newly terminated link and the current average duration. It is interesting to study how this change may affect the accuracy of the estimated link duration and hence the IPD value of paths.

Third, we want to evaluate the accuracy of the exponential distribution approximation and the expected path duration estimation (IPD value), and its impact on the performance of the path selection scheme. The performance of the local path recovery scheme proposed in Section 6.2 clearly relies on a reliable estimation of the IPD value.

Appendix A

Mobility models used for research in MANETs

One of the important methods in studying MANETs is through simulation. Simulations not only help to evaluate the performance of various protocols proposed for different type of networks, but also may reveal problems that have not been noticed or understood through analytical studies. Among the components of these simulation settings, the mobility model each node adopts plays an important role. As discussed in Chapter 2, the performance of routing protocols are closely related to the mobility model of the nodes. Hence, there is much interest to model the mobility patterns of nodes[1] in real communication networks. There are several survey papers on this mobility modelling subject [7, 19, 20]. Commonly used

[1]In real network, the notion of nodes can refer to people, vehicles, sensors and etc.

mobility models include: Random Walk Model, Random Waypoint Model, Group Mobility Model (Reference Point Group Mobility Model, Pursue Mobility Model, etc.), Freeway Model, Manhattan Model and etc. In the following, we describe several of them in more details.

A.1 Random Walk mobility model

The Random Walk Mobility Model was used to model nodes moving in extremely unpredictable ways. Under this mobility model, a node moves in a randomly chosen direction and speed uniformly distributed between $[s_{\min}, s_{\max}]$. Each node changes its movement, both in direction and speed, either per time slot or per distance unit. When a node reaches the domain boundary, it bounces off with an angle determined by the incoming direction, and continues along this new direction. The domain in which nodes move can be of 1-D, 2-D or higher dimension. For random walk model in 1-D and 2-D, the nodes can return to the origin of the center of the network with probability one. This model is also referred to as Browning Motion model.

A.2 Random Waypoint mobility model

Random Waypoint (RWP) mobility is a commonly adopted model for studying MANETs. Under this model, nodes are initially distributed uniformly on a domain. Each node selects a random waypoint uniformly distributed on the domain as its next destination. It then moves along a straight line to this destination using a constant speed uniformly chosen from a speed range s_{min} and s_{max}. After reaching the destination, it then repeats the whole process again, either with or without idling at the destination. If the node idles before it takes off to the next waypoint, then this model is called the RWP model with idling. As the network runs, the steady state is reached, under which the distribution of the location of the nodes and the distribution of their speeds converge to their limiting distributions, respectively. Many analytical and numerical approaches for studying these limiting distributions are available, *e.g.,* in [4, 5, 15, 25, 29, 48].

A.3 Reference Point Group mobility model

Under Reference Point Group (RPG) mobility model, introduced in [16], nodes are divided into groups. In each group, one of them is selected as the group leader. The leaders can move freely on the domain of the network. Other nodes must follow their leaders by always staying in a circular area centered at the leader. To be specific, when a leader selects its destination, each node will pick up a reference point that is located within certain radius of this destination point, and move towards this point along a straight line. This model can be used to model many real situations. For example, in battle fields, soldiers move in group. In disaster rescue operations, different teams of personnel may have different purpose and hence move together in a common routine.

A.4 Manhattan Model and Freeway Model

Under Manhattan Model and Freeway Model, nodes' movements are restricted on a pre-determined map. The Manhattan model is used to model mobility of people in an urban area. The map on which nodes move has a grid structure. This grid structure models the streets in a city. At the cross points of the grid (intersections of streets), a node is free to change its direction. The Freeway model is for modelling cars moving along the lanes on a highway. Under this model, the movement of nodes is constrained both by the lane it stays and by the nodes moving ahead of it in the same lane.

Appendix B

Proofs

B.1 Proof of Theorem 5.1

The proof of this theorem directly uses the theorem in [44]. The theorem in [44, p. 798] is on the distributional convergence of the maximum of a sequence of stationary sequence. In our problem, we intend to study their minimum. Thus, we introduce an array of sequences of rvs, $\{W_\ell^{(n)}, \ell = 1, \ldots, h(n), n = 1, 2, \ldots\}$, defined as $W_\ell^{(n)} = (X_\ell^{(n)})^{-1}$, for $\ell = 1, \ldots, h(n)$, where $h(n)$ increases with n. The maximum of each sequence $\{W_\ell^{(n)}, \ell = 1, \ldots, h(n)\}$ is denoted by $M_{h(n)}$.

The sequence $\{W_\ell^{(n)}\}$ is also stationary and m-dependent. Moreover, the condition in (5.2) can be written as

$$\lim_{c\to\infty} \frac{1}{\mathbf{P}\left[W_\ell^{(n)} > c\right]} \max_{|\ell-\ell'|\leq m} \mathbf{P}\left[W_\ell^{(n)} > c, W_{\ell'}^{(n)} > c\right] = 0 \ .$$

Theorem in [44] tells us that if there exists a sequence $c_n(x)$ and a constant $\lambda > 0$, such that $h(n) \cdot \mathbf{P}\left[W_\ell^{(n)} > c_n(x)\right] \to \lambda x$, then $\mathbf{P}\left[M_{h(n)} \leq c_n(x)\right]$ converges to $\exp(-\lambda x)$.

In the context of our problem, for any fix $x > 0$, we let $c_n(x) = \frac{h(n)}{x}, n = 1, 2, \ldots$. Notice that

$$\lim_{n\to\infty} h(n) \cdot \mathbf{P}\left[W_\ell^{(n)} > \frac{h(n)}{x}\right] = \lim_{n\to\infty} h(n) \cdot \mathbf{P}\left[X_\ell^{(n)} < \frac{x}{h(n)}\right] .$$

We also have the relation

$$\mathbf{P}\left[X_\ell^{(n)} < x\right] = \frac{1}{m(G)} \int_0^x (1 - G(y))\, dy$$

from (3.9), where $m(G)$ is the mean of the link duration. It then follows that

$$\begin{aligned} \lim_{n\to\infty} h(n) \cdot \mathbf{P}\left[W_\ell^{(n)} > \frac{h(n)}{x}\right] &= \lim_{n\to\infty} \frac{n}{m(G)} \left(\frac{x}{h(n)} + o\left(\frac{1}{h(n)}\right)\right) \\ &= \frac{1}{m(G)} x . \end{aligned}$$

Here $\lambda = (m(G))^{-1}$. Therefore, the conditions for the theorem in [44, p. 798] are satisfied by the array of sequences $\{W_\ell^{(n)}\}$.

The proof of theorem 5.1 is now one step away. Under the notation defined in Chapter 4, $H(n)$ is equivalent to the $h(n)$ here. We now have

$$\begin{aligned} \lim_{n\to\infty} \mathbf{P}\left[h(n) \cdot Z^{(n)} > x\right] &= \lim_{n\to\infty} \mathbf{P}\left[Z^{(n)} \leq \frac{x}{h(n)}\right] \\ &= \lim_{n\to\infty} \mathbf{P}\left[M_{h(n)} \leq c_n(x)\right] \\ &= \exp(-\lambda x) \\ &= \exp(-\frac{x}{m(G)}) . \end{aligned}$$

This completes the proof.

B.2 Proof of Theorem 5.2

To prove the theorem, we show that for any $x \in (0, \infty)$,

$$\lim_{n\to\infty} \mathbf{P}\left[H(n) \cdot Z^{(n)} > x\right] = \exp\left(-\lambda x\right) \ . \tag{B.1}$$

Same as in the proof of Theorem 5.1, we introduce an array $\{W_\ell^{(n)}\}$, $\ell = 1, \ldots, H(n)$, $n = 1, 2, \ldots$, where $W_\ell^{(n)} = (X_\ell^{(n)})^{-1}$. Thus, equation (B.1) is equivalent to

$$\lim_{n\to\infty} \mathbf{P}\left[\max_{\ell=1,\ldots,H(n)} W_\ell^{(n)} \le \frac{H(n)}{x}\right] = \exp\left(-\lambda x\right) \ . \tag{B.2}$$

We first introduce some notation. Let E be an interval, as defined in Section 5.2.1. Let $M^{(n)}(E)$ denote $\max_{j\in E}\{W_j^{(n)}\}$. If $E = \{j_1, \ldots, j_2\}$ and $E' = \{j_1', \ldots, j_2'\}$ are two intervals with $j_1' > j_2$, we say that E and E' are separated by $j_1' - j_2$.

Suppose that we have a sequence $\{k(n), n = 1, 2, \ldots\}$ such that $\lim_{n\to\infty} k(n) = \infty$ and $\lim_{n\to\infty} \frac{k(n)}{H(n)} = 0$. For each $n = 1, 2, \ldots$, we divide the set $\{1, \ldots, H(n)\}$ into $k(n) + 1$ consecutive intervals, $I_{k(n),j}^{(n)}$, $j = 1, \ldots, k(n) + 1$, same as done in Section 5.2.1. Furthermore, let $\{m(n), n = 1, 2, \ldots\}$ be a sequence of integers such that $k(n) < m(n) < n'$, where $n' := \lfloor H(n)/k(n) \rfloor$. The sequences $\{k(n)\}$ and $\{m(n)\}$ also satisfy the following conditions:

(i) $\lim_{n\to\infty} m(n) = \infty$,

(ii) $\lim_{n\to\infty} \frac{k(n)}{m(n)} = 0$,

(iii) $\lim_{n\to\infty} \frac{m(n)}{H(n)} = 0$,

(iv) $\lim_{n\to\infty} \frac{m(n)\cdot k(n)}{H(n)} = 0$.

When there is no confusion, we write k and m instead of $k(n)$ and $m(n)$ respectively. For each interval $I_{k,j}^{(n)}$, $j = 1, \ldots, k$, we divide it into two subintervals $\underline{I}_{k,j}^{(n)}$ and $\bar{I}_{k,j}^{(n)}$ of length $n' - m$ and m, respectively; *i.e.*,

$$\underline{I}_{k,j}^{(n)} = \{(j-1)\cdot n' + 1, (j-1)\cdot n' + 2, \ldots, j\cdot n' - m\}$$

and

$$\bar{I}_{k,j}^{(n)} = \{j\cdot n' - m + 1, \ldots, j\cdot n'\} \ .$$

We denote $M^{(n)}(I_{k,j}^{(n)})$, $j = 1, \ldots, k$, by $M_{k,j}^{(n)}$ for notational convenience.

In order to prove (B.2), we first show that

$$\lim_{n\to\infty} \prod_{j=1}^{k(n)} \mathbf{P}\left[M_{k(n),j}^{(n)} \leq u_n(x)\right] = \exp(-\lambda x) \ , \tag{B.3}$$

then we show that the difference between the left hand sides of (B.2) and (B.3) goes away as n increases, *i.e.*,

$$\left|\mathbf{P}\left[\max_{\ell=1,\ldots,H^{(n)}} W_\ell^{(n)} \leq u_n(x)\right] - \prod_{j=1}^{k(n)} \mathbf{P}\left[M_{k(n),j}^{(n)} \leq u_n(x)\right]\right| \to 0 \ . \tag{B.4}$$

The prove of (B.3) is established using the fact that both the upper and lower bounds of the left side of (B.3) converge to $\exp(-\lambda x)$.

The event $\{M_{k,j}^{(n)} > u_n\}$ can be written as $\bigcup_{i \in I_{k,j}^{(n)}} \{W_i^{(n)} > u_n\}$ for $j = 1, \ldots, k$. Hence, we can write the upper and lower bound of $\mathbf{P}\left[M_{k,j}^{(n)} > u_n\right]$ as

$$\begin{aligned}
&\sum_{i \in I_{k,j}^{(n)}} \mathbf{P}\left[W_i^{(n)} > u_n\right] - \sum_{i,i' \in I_{k,j}^{(n)}, i \neq i'} \mathbf{P}\left[W_i^{(n)} > u_n, W_{i'}^{(n)} > u_n\right] \\
&\leq \ \mathbf{P}\left[M_{k,j}^{(n)} > u_n\right] \\
&\leq \ \sum_{i \in I_{k,j}^{(n)}} \mathbf{P}\left[W_i^{(n)} > u_n\right] \ .
\end{aligned} \tag{B.5}$$

Using (B.5) we can derive the bounds for $\prod_{j=1}^{k(n)} \mathbf{P}\left[M_{k(n),j}^{(n)} \leq u_n(x)\right]$ as:

$$\begin{aligned}
&\prod_{j=1}^{k} \left(1 - \sum_{i \in I_{k,j}^{(n)}} \mathbf{P}\left[W_i^{(n)} > u_n\right]\right) \\
&\leq \prod_{j=1}^{k} \left(\mathbf{P}\left[M_{k,j}^{(n)} \leq u_n\right]\right) \\
&\leq \prod_{j=1}^{k} \left(1 - \sum_{i \in I_{k,j}^{(n)}} \mathbf{P}\left[W_i^{(n)} > u_n\right] + \sum_{i,i' \in I_{k,j}^{(n)}, i \neq i'} \mathbf{P}\left[W_i^{(n)} > u_n, W_{i'}^{(n)} > u_n\right]\right) \ .
\end{aligned} \tag{B.6}$$

Take the sequence $u_n = \frac{H(n)}{x}$, we can show that for fixed x,

$$\lim_{n \to \infty} \sum_{i \in I_{k(n),j}^{(n)}} \mathbf{P}\left[W_i^{(n)} > u_n\right] = 0 \ . \tag{B.7}$$

and

$$\lim_{n \to \infty} \sum_{j=1}^{k(n)} \sum_{i \in I_{k(n),j}^{(n)}} \mathbf{P}\left[W_i^{(n)} > u_n\right] = \lambda \cdot x \ . \tag{B.8}$$

The argument for (B.7) follows from Assumption 5.2. Using the relations

$$\mathbf{P}\left[X_\ell^{(n)} < x\right] = \frac{1}{m(G^{(n)})} \int_0^x (1 - G(y)) \ dy \ , \ W_\ell^{(n)} = (X_\ell^{(n)})^{-1} \ \text{, and } u_n = \frac{H(n)}{x} \ ,$$

we have

$$\begin{aligned}\sum_{i\in I^{(n)}_{k(n),j}} \mathbf{P}\left[W_i^{(n)} > u_n\right] &= \sum_{i\in I^{(n)}_{k(n),j}} \lambda_i^{(n)} \int_0^{\frac{x}{H(n)}} (1-G_i^{(n)}(y))dy \\ &\leq \sum_{i\in I^{(n)}_{k(n),j}} \lambda_i \frac{x}{H(n)} \ .\end{aligned}$$

Then, under Assumption 5.2, as n increases, we have

$$\sum_{i\in I^{(n)}_{k(n),j}} \lambda_i \frac{x}{H(n)} = x\cdot O\left(\frac{|I^{(n)}_{k(n),j}|}{H(n)}\right) \ .$$

Since the length of interval $I^{(n)}_{k(n),j}$ is bounded by $\frac{H(n)}{k(n)}$, we have

$$x\cdot O\left(\frac{|I^{(n)}_{k(n),j}|}{H(n)}\right) = x\cdot O\left(\frac{1}{k(n)}\right) \ .$$

Since $\lim_{n\to\infty} k(n) = \infty$ and x is some fixed positive real number, equation (B.7) follows.

To prove (B.8), we notice that

$$\sum_{j=1}^{k(n)} \sum_{i\in I^{(n)}_{k(n),j}} \mathbf{P}\left[W_i^{(n)} > u_n\right] = \sum_{i=1}^{H(n)} \mathbf{P}\left[W_i^{(n)} > u_n\right] - \sum_{i\in I^{(n)}_{k(n),k(n)+1}} \mathbf{P}\left[W_i^{(n)} > u_n\right] \quad \text{(B.9)}$$

The first term on the right hand side of (B.9) converges to λx as $n\to\infty$ because of Assumption 4.1 [1] and the second term converges to zero from Assumption 5.2.

From (B.7) - (B.8), we have

$$\lim_{n\to\infty} \prod_{j=1}^{k(n)} \left(1 - \sum_{i\in I^{(n)}_{k(n),j}} \mathbf{P}\left[W_i^{(n)} > u_n\right]\right) = \exp(-\lambda x) \ .$$

[1] Assumption 4.1 is equivalent to $\lim_{n\to\infty} \sum_{i=1}^{H(n)} \mathbf{P}\left[W_i^{(n)} > u_n\right] = \lambda x$ using the relations

$$\mathbf{P}\left[X_\ell^{(n)} < x\right] = \frac{1}{m(G^{(n)})}\int_0^x (1-G(y))\ dy\ ,\ W_\ell^{(n)} = (X_\ell^{(n)})^{-1}\ ,\text{and } u_n = \frac{H(n)}{x}\ .$$

Moreover, using the relation

$$\sum_{i,i' \in I_{k(n),j}^{(n)}, i \neq i'} \mathbf{P}\left[W_i^{(n)} > u_n, W_{i'}^{(n)} > u_n\right] = o\left(\frac{1}{k(n)}\right)$$

from condition $D'(u_n = \frac{H(n)}{x})$, we have

$$\begin{aligned}
&\lim_{n\to\infty} \prod_{j=1}^{k(n)} \left(1 - \sum_{i \in I_{k(n),j}^{(n)}} \mathbf{P}\left[W_i^{(n)} > u_n\right]\right. \\
&\qquad \left. + \sum_{i,i' \in I_{k(n),j}^{(n)}, i \neq i'} \mathbf{P}\left[W_i^{(n)} > u_n, W_{i'}^{(n)} > u_n\right]\right) \\
&= \exp(-\lambda x) \ .
\end{aligned}$$

Therefore, take the limit of the lower and upper bounds of $\prod_{j=1}^{k(n)} \left(\mathbf{P}\left[M_{k(n),j}^{(n)} \leq u_n\right]\right)$ in (B.6), we obtain

$$\lim_{n\to\infty} \prod_{j=1}^{k(n)} \left(\mathbf{P}\left[M_{k(n),j}^{(n)} \leq u_n\right]\right) = \exp(-\lambda x) \ . \tag{B.10}$$

The last step is to show (B.4). This is given by he following lemma. Its proof is provided in Appendix B.3.

Lemma B.1 *Suppose the sequence* $\{m(n), n \geq 1\}$ *satisfies the condition* $D(u_n = \frac{H(n)}{x})$. *Furthermore, sequence* $\{m(n)\}$ *and* $\{k(n)\}$ *satisfy conditions* $(i) - (iv)$. *In addition,* $\lim_{n\to\infty} k(n) \cdot \alpha_{n,m(n)} = 0$. *Then, we have*

$$\lim_{n\to\infty} \left| \mathbf{P}\left[M_n \leq u_n\right] - \prod_{j=1}^{k(n)} \mathbf{P}\left[M_{k(n),j}^{(n)} \leq u_n\right] \right| = 0 \ ,$$

where $M_n := \max(W_1^{(n)}, \ldots, W_{H(n)}^{(n)})$.

Combining (B.10) and Lemma B.1 we conclude that

$$\begin{aligned}
\lim_{n\to\infty} \mathbf{P}\left[H(n)\cdot Z^{(n)} > x\right] &= \lim_{n\to\infty} \mathbf{P}\left[\max_{\ell=1,\ldots,H(n)} W_\ell^{(n)} \le \frac{H(n)}{x}\right] \\
&= \lim_{n\to\infty} \mathbf{P}\left[M_n \le u_n\right] \\
&= \lim_{n\to\infty} \prod_{j=1}^{k(n)} \mathbf{P}\left[M_{k(n),j}^{(n)} \le u_n\right] \\
&= \exp(-\lambda x)\ .
\end{aligned}$$

This completes the proof.

B.3 Proofs of Lemma B.1

We first introduce some auxiliary results to be used.

Lemma B.2 *Suppose that condition $D(u_n)$ holds for some sequence $\{u_n, n = 1, 2, \ldots\}$. Let n, r, and m be fixed positive integers and $E_1, \ldots, E_r$ subintervals of $\{1, \ldots, H(n)\}$ such that any two E_i and E_j, $i \ne j$, are separated by at least m. Then, we have*

$$\left|\mathbf{P}\left[\bigcap_{j=1}^{r}\{M^{(n)}(E_j) \le u_n\}\right] - \prod_{j=1}^{r} \mathbf{P}\left[M^{(n)}(E_j) \le u_n\right]\right| \le (r-1)\cdot \alpha_{n,m}\ .$$

Proof: For notational convenience, we write $A_j^{(n)} = \{M^{(n)}(E_j) \le u_n\}$. Let $E_j = \{k_j, \ldots, l_j\}$, where $k_1 \le l_1 < k_2 \le \ldots \le l_r$. Then, under condition $D(u_n)$, we get

$$\begin{aligned}
\left|\mathbf{P}\left[A_1^{(n)} \cap A_2^{(n)}\right] - \mathbf{P}\left[A_1^{(n)}\right]\mathbf{P}\left[A_2^{(n)}\right]\right| &= \left|\mathbf{J}^{(n)}_{k_1\ldots l_1, k_2\ldots l_2}(u_n) - \mathbf{J}^{(n)}_{k_1\ldots l_1}(u_n)\mathbf{J}^{(n)}_{k_2\ldots l_2}(u_n)\right| \\
&\le \alpha_{n,m}
\end{aligned}$$

since $k_2 - l_1 \geq m$. Similarly,

$$\begin{aligned}
&\left|\mathbf{P}\left[A_1^{(n)} \cap A_2^{(n)} \cap A_3^{(n)}\right] - \mathbf{P}\left[A_1^{(n)}\right]\mathbf{P}\left[A_2^{(n)}\right]\mathbf{P}\left[A_3^{(n)}\right]\right| \\
&\leq \left|\mathbf{P}\left[A_1^{(n)} \cap A_2^{(n)} \cap A_3^{(n)}\right] - \mathbf{P}\left[A_1^{(n)} \cap A_2^{(n)}\right]\mathbf{P}\left[A_3^{(n)}\right]\right| \\
&\qquad + \left|\mathbf{P}\left[A_1^{(n)} \cap A_2^{(n)}\right] - \mathbf{P}\left[A_1^{(n)}\right]\mathbf{P}\left[A_2\right]\right| \cdot \mathbf{P}\left[A_3\right] \\
&\leq 2\ \alpha_{n,m}
\end{aligned}$$

since $E_1 \cup E_2 \subseteq \{k_1, \ldots, l_2\}$ and $k_3 - l_2 \geq m$. By applying the same argument repeatedly, the lemma follows. ∎

Lemma B.3 *Suppose that the condition $D(u_n)$ holds. For any fixed k, the following statements hold:*

(i)

$$\begin{aligned}
0 &\leq \mathbf{P}\left[\bigcap_{j=1}^{k}\{M^{(n)}(\underline{I}_{k,j}^{(n)}) \leq u_n\}\right] - \mathbf{P}\left[M_n \leq u_n\right] \\
&\leq \sum_{j=1}^{k}\mathbf{P}\left[M^{(n)}(\underline{I}_{k,j}^{(n)}) \leq u_n < M^{(n)}(\bar{I}_{k,j}^{(n)})\right] \\
&\quad + \mathbf{P}\left[u_n < M^{(n)}(I_{k,k+1}^{(n)})\right]\ ,
\end{aligned} \tag{B.11}$$

(ii)

$$\begin{aligned}
&\left|\mathbf{P}\left[\bigcap_{j=1}^{k}\left\{M^{(n)}(\underline{I}_{k,j}^{(n)}) \leq u_n\right\}\right] - \prod_{j=1}^{k}\mathbf{P}\left[M^{(n)}(\underline{I}_{k,j}^{(n)}) \leq u_n\right]\right| \\
&\leq (k-1)\cdot\alpha_{n,m}\ ,
\end{aligned}$$

(iii)

$$0 \le \prod_{j=1}^{k} \mathbf{P}\left[M^{(n)}(\underline{I}_{k,j}^{(n)}) \le u_n\right] - \prod_{j=1}^{k} \mathbf{P}\left[M_{k,j}^{(n)} \le u_n\right]$$
$$\le \prod_{j=1}^{k} \left(1 + \mathbf{P}\left[M^{(n)}(\underline{I}_{k,j}^{(n)}) \le u_n < M^{(n)}(\bar{I}_{k,j}^{(n)})\right]\right) - 1 \ .$$

Proof: Claim (i) follows from the observation that

$$\{M_n \le u_n\} \subset \bigcap_{j=1}^{k} \{M^{(n)}(\underline{I}_{k,j}^{(n)}) \le u_n\}$$

and their difference is given by the event

$$\left(\bigcup_{j=1}^{k} \left\{M^{(n)}(\underline{I}_{k,j}^{(n)}) \le u_n < M^{(n)}(\bar{I}_{k,j}^{(n)})\right\}\right) \bigcup \{u_n < M^{(n)}(I_{k,k+1}^{(n)})\} \ ,$$

whose probability can be bounded using the union bound in (B.11).

Claim (ii) follows directly from Lemma B.2 by replacing E_j with $\underline{I}_{k,j}^{(n)}$.

In order to prove claim (iii), we first note that, for $j = 1, \dots, k$,

$$\mathbf{P}\left[M^{(n)}(\underline{I}_{k,j}^{(n)}) \le u_n\right] - \mathbf{P}\left[M_{k,j}^{(n)} \le u_n\right] = \mathbf{P}\left[M^{(n)}(\underline{I}_{k,j}^{(n)}) \le u_n < M^{(n)}(\bar{I}_{k,j}^{(n)})\right] \ .$$

Hence

$$\prod_{j=1}^{k} \mathbf{P}\left[M^{(n)}(\underline{I}_{k,j}^{(n)}) \le u_n\right] - \prod_{j=1}^{k} \mathbf{P}\left[M_{k,j}^{(n)} \le u_n\right]$$
$$= \prod_{j=1}^{k} \left(\mathbf{P}\left[M_{k,j}^{(n)} \le u_n\right] + \mathbf{P}\left[M^{(n)}(\underline{I}_{k,j}^{(n)}) \le u_n < M^{(n)}(\bar{I}_{k,j}^{(n)})\right]\right)$$
$$- \prod_{j=1}^{k} \mathbf{P}\left[M_{k,j}^{(n)} \le u_n\right] \ . \tag{B.12}$$

Expand (B.12) directly and the term $\prod_{j=1}^{k} \mathbf{P}\left[M_{k,j}^{(n)} \le u_n\right]$ will be cancelled out. Then since $\mathbf{P}\left[M_{k,j}^{(n)} \le u_n\right] \le 1$ for all k and j, we have (B.12) being upper bounded by

$$\prod_{j=1}^{k}\left(1+\mathbf{P}\left[M^{(n)}(\underline{I}_{k,j}^{(n)}) \leq u_n < M^{(n)}(\bar{I}_{k,j}^{(n)})\right]\right) - 1 \ .$$

$\blacksquare$

Lemma B.4 *Under Assumption 5.2 and condition* $D(u_n = \frac{H(n)}{x})$ *with sequence* $m(n)$ *and* $k(n)$ *satisfying the conditions* $(i)-(iv)$ *in Appendix B.2,*

$$\mathbf{P}\left[M^{(n)}(\underline{I}_{k(n),j}^{(n)}) \leq u_n < M^{(n)}(\bar{I}_{k(n),j}^{(n)})\right] = o\left(\frac{1}{k(n)}\right) \ ,$$

for all $j = 1, \ldots, k(n)$.

Proof: It is straightforward that

$$\begin{aligned}
&\mathbf{P}\left[M^{(n)}(\underline{I}_{k(n),j}^{(n)}) \leq u_n < M^{(n)}(\bar{I}_{k(n),j}^{(n)})\right] \\
&= \mathbf{P}\left[\left(\bigcap_{i\in\underline{I}_{k(n),j}^{(n)}} \{W_i^{(n)} \leq u_n\}\right) \bigcap \left(\bigcup_{i\in\bar{I}_{k(n),j}^{(n)}} \{W_i^{(n)} > u_n\}\right)\right] \\
&\leq \sum_{i\in\bar{I}_{k(n),j}^{(n)}} \mathbf{P}\left[W_i^{(n)} > u_n\right] \ .
\end{aligned}$$

Then, under Assumption 5.2, as n increases,

$$\sum_{i\in\bar{I}_{k(n),j}^{(n)}} \mathbf{P}\left[W_i^{(n)} > u_n\right] \;=\; x\cdot O\left(\frac{|\bar{I}_{k(n),j}^{(n)}|}{H(n)}\right)$$

Since x is fixed and hence can be absorbed in $O(\cdot)$, we have

$$\begin{aligned}
x\cdot O\left(\frac{|\bar{I}_{k(n),j}^{(n)}|}{H(n)}\right) &= \frac{1}{k(n)}\cdot O\left(\frac{m(n)\cdot k(n)}{H(n)}\right) \\
&= o\left(\frac{1}{k(n)}\right) \ .
\end{aligned}$$

The last equality follows from the condition $m(n) \cdot k(n) = o(H(n))$. ■

We now proceed with the proof of Lemma B.1. Combining claims (i) - (iii) in Lemma B.3, we have

$$\begin{aligned}
&\left| \mathbf{P}\left[M_n \le u_n\right] - \prod_{j=1}^{k(n)} \mathbf{P}\left[M^{(n)}_{k(n),j} \le u_n\right] \right| \\
&\le \left| \mathbf{P}\left[M_n \le u_n\right] - \prod_{j=1}^{k(n)} \mathbf{P}\left[\bigcap_{j=1}^{k(n)} \left\{ M^{(n)}(\underline{I}^{(n)}_{k(n),j}) \le u_n \right\}\right] \right| \\
&\quad + \left| \mathbf{P}\left[\bigcap_{j=1}^{k(n)} \left\{ M^{(n)}(\underline{I}^{(n)}_{k(n),j}) \le u_n \right\}\right] - \prod_{j=1}^{k(n)} \mathbf{P}\left[M^{(n)}(\underline{I}^{(n)}_{k(n),j}) \le u_n\right] \right| \\
&\quad + \left| \prod_{j=1}^{k(n)} \mathbf{P}\left[M^{(n)}(\underline{I}^{(n)}_{k(n),j}) \le u_n\right] - \prod_{j=1}^{k(n)} \mathbf{P}\left[M^{(n)}_{k(n),j} \le u_n\right] \right| \\
&\le \sum_{j=1}^{k(n)} \mathbf{P}\left[M^{(n)}(\underline{I}^{(n)}_{k(n),j}) \le u_n < M^{(n)}(\bar{I}^{(n)}_{k(n),j})\right] \\
&\quad + \mathbf{P}\left[u_n < M^{(n)}(I^{(n)}_{k(n),k(n)+1})\right] + (k(n)-1) \cdot \alpha_{n,m(n)} \\
&\quad + \prod_{j=1}^{k(n)} \left(1 + \mathbf{P}\left[M^{(n)}(\underline{I}^{(n)}_{k(n),j}) \le u_n < M^{(n)}(\bar{I}^{(n)}_{k(n),j})\right] \right) - 1 \, . \qquad \text{(B.13)}
\end{aligned}$$

First, Lemma B.4 tells us that

$$\begin{aligned}
&\lim_{n\to\infty} \sum_{j=1}^{k(n)} \mathbf{P}\left[M^{(n)}(\underline{I}^{(n)}_{k(n),j}) \le u_n < M^{(n)}(\bar{I}^{(n)}_{k(n),j})\right] \\
&= \lim_{n\to\infty} \sum_{j=1}^{k(n)} o\left(\frac{1}{k(n)}\right) \\
&= 0 \, . \qquad \text{(B.14)}
\end{aligned}$$

Follow the same argument,

$$
\begin{aligned}
&\lim_{n\to\infty}\prod_{j=1}^{k(n)}\left(1+\mathbf{P}\left[M^{(n)}(\underline{I}^{(n)}_{k(n),j})\le u_n < M^{(n)}(\bar{I}^{(n)}_{k(n),j})\right]\right)\\
&=\lim_{n\to\infty}\prod_{j=1}^{k(n)}\left(1+o\left(\frac{1}{k(n)}\right)\right)\\
&=1\ . \qquad\qquad (\text{B.15})
\end{aligned}
$$

Then, since

$$
\limsup_{n\to\infty}\frac{|I^{(n)}_{k(n)+1}|}{H(n)}\le\limsup_{n\to\infty}\frac{k(n)-1}{H(n)}=0\ ,
$$

it follows from Assumption 5.2 that

$$
\begin{aligned}
\mathbf{P}\left[u_n < M^{(n)}(I^{(n)}_{k(n),k(n)+1})\right] &= \mathbf{P}\left[\bigcup_{i\in I^{(n)}_{k(n),k(n)+1}}\left\{W_i^{(n)}>u_n\right\}\right]\\
&\le \sum_{i\in I^{(n)}_{k(n),k(n)+1}}\mathbf{P}\left[W_i^{(n)}>u_n\right]\ . \qquad (\text{B.16})
\end{aligned}
$$

As n increases, (B.16) goes to 0.

From (B.14) - (B.16), along with the assumption $\lim_{n\to\infty} k(n)\cdot\alpha_{n,m(n)} = 0$, the right hand side of (B.13) goes to 0 as $n\to\infty$. This completes the proof of Lemma B.1.

B.4 Proof of Lemma 5.1

To prove this lemma, we need to prove the existence of an array $A=\{\alpha_{n,m(n)}\}$ that satisfy the condition $D(u_n)$ for array $\{W_\ell^{(n)}\}$. Here, we prove a stronger statement that for any integers

$$1 \leq i_1 < \cdots < i_p < j_1 < \cdots < j_q \leq H(n) \ ,$$

we have

$$\left| \mathbf{J}^{(n)}_{i_1 \ldots i_p j_1 \ldots j_q}(u_n) - \mathbf{J}^{(n)}_{i_1 \ldots i_p}(u_n) \mathbf{J}^{(n)}_{j_1 \ldots j_q}(u_n) \right| = 0 \ . \tag{B.17}$$

To simplify the notation, we introduce two events $\mathcal{E}_1^{(n)}$ and $\mathcal{E}_2^{(n)}$ as follows:

$$\mathcal{E}_1^{(n)} := \{ W_{i_1}^{(n)} \leq \frac{H(n)}{x}, \ldots, W_{i_p}^{(n)} \leq \frac{H(n)}{x} \}$$

and

$$\mathcal{E}_2^{(n)} := \{ W_{j_1}^{(n)} \leq \frac{H(n)}{x}, \ldots, W_{j_q}^{(n)} \leq \frac{H(n)}{x} \}$$

Since for any two events A and B, we have

$$\begin{aligned} \mathbf{P}[A \cap B] &= \mathbf{P}[A] + \mathbf{P}[B] - \mathbf{P}[A \cup B] \\ &= 1 - \mathbf{P}\left[\bar{A}\right] + 1 - \mathbf{P}\left[\bar{B}\right] - \mathbf{P}[A \cup B] \\ &\geq 1 - \mathbf{P}\left[\bar{A}\right] - \mathbf{P}\left[\bar{B}\right] \ . \end{aligned}$$

It then follows that

$$\mathbf{P}\left[\mathcal{E}_1^{(n)}\right] \geq 1 - \sum_{k=1}^{p} \mathbf{P}\left[W_{i_k}^{(n)} > \frac{H(n)}{x}\right]$$
$$\mathbf{P}\left[\mathcal{E}_2^{(n)}\right] \geq 1 - \sum_{k=1}^{q} \mathbf{P}\left[W_{j_k}^{(n)} > \frac{H(n)}{x}\right]$$

and

$$\mathbf{P}\left[\mathcal{E}_1^{(n)} \cap \mathcal{E}_2^{(n)}\right] \geq 1 - \sum_{k=1}^{p} \mathbf{P}\left[W_{i_k}^{(n)} > \frac{H(n)}{x}\right] - \sum_{k=1}^{q} \mathbf{P}\left[W_{j_k}^{(n)} > \frac{H(n)}{x}\right] \ .$$

Therefore, we can calculate upper and lower bounds on

$\mathbf{J}^{(n)}_{i_1\ldots i_p j_1\ldots j_q}(u_n) - \mathbf{J}^{(n)}_{i_1\ldots i_p}(u_n)\mathbf{J}^{(n)}_{j_1\ldots j_q}(u_n)$ as follows:

$$\begin{aligned}
&-\sum_{k=1}^{p}\mathbf{P}\left[W^{(n)}_{i_k} > \frac{H(n)}{x}\right] - \sum_{k=1}^{q}\mathbf{P}\left[W^{(n)}_{j_k} > \frac{H(n)}{x}\right] \\
\leq\ & \mathbf{J}^{(n)}_{i_1\ldots i_p j_1\ldots j_q}(u_n) - \mathbf{J}^{(n)}_{i_1\ldots i_p}(u_n)\mathbf{J}^{(n)}_{j_1\ldots j_q}(u_n) \\
\leq\ & \sum_{k=1}^{p}\mathbf{P}\left[W^{(n)}_{i_k} > \frac{H(n)}{x}\right] + \sum_{k=1}^{q}\mathbf{P}\left[W^{(n)}_{j_k} > \frac{H(n)}{x}\right] \\
& - \left(\sum_{k=1}^{p}\mathbf{P}\left[W^{(n)}_{i_k} > \frac{H(n)}{x}\right]\right)\left(\sum_{k=1}^{q}\mathbf{P}\left[W^{(n)}_{j_k} > \frac{H(n)}{x}\right]\right) .
\end{aligned}$$

Since both $\sum_{k=1}^{p}\mathbf{P}\left[W^{(n)}_{i_k} > \frac{H(n)}{x}\right] \to 0$ and $\sum_{k=1}^{q}\mathbf{P}\left[W^{(n)}_{j_k} > \frac{H(n)}{x}\right] \to 0$ from Assumption 5.2, the upper and lower bounds on $\mathbf{J}^{(n)}_{i_1\ldots i_p j_1\ldots j_q}(u_n) - \mathbf{J}^{(n)}_{i_1\ldots i_p}(u_n)\mathbf{J}^{(n)}_{j_1\ldots j_q}(u_n)$ converge to 0 as $n \to \infty$ and (B.17) follows.

B.5 Proof of threshold updating

The proof of the convergence of sequence $\{\gamma_n, n \geq 0\}$ can be found in the literature of stochastic approximation theory [6, 22, 38]. As defined in Section 6.2.1, the indicator function Q_n ($Q_n = 1$ when the $(n+1)$-th trial of the backup path is successful and 0 otherwise, given the current threshold γ_n), is a Bernoulli rv with an unknown distribution $H(\cdot)$. We use $m(\gamma)$ to denote its mean. In other words, $m(\gamma)$ is the probability a backup path is available when tried using the threshold equals to γ. We assume that there exists a γ^* such that $m(\gamma^*) = P_{target}$. Thus, our problem is to find a γ^* such that $m(\gamma^*) = P_{target}$.

Recall the iteration of thresholds updating given in (6.2):

$$\gamma_{n+1} = \gamma_n + \epsilon_n (P_{target} - Q_n) \ . \tag{B.18}$$

It can be written as

$$\gamma_{n+1} = \gamma_n + \epsilon_n (P_{target} - m(\gamma_n)) + \epsilon_n (m(\gamma_n) - Q_n) \ . \tag{B.19}$$

Let $\mathcal{F}_n$ be a sequence of σ-algebras that each $\mathcal{F}_n$ is generated by $\{(Q_i, \gamma_i), i = 0, \ldots, n-1, \gamma_n\}$. From the definition of Q_n, $\mathbf{E}[Q_n|\mathcal{F}_n] = m(\gamma_n)$. Hence the sequence $\{m(\gamma_n) - Q_n, n \geq 1\}$ forms a Martingale difference with respect to σ-algebra $\mathcal{F}_n$.

The following lemma is used to complete the proof of the convergence of the sequence γ_n, which can be found in [6].

Lemma B.5 *(i) If the sequence of the step sizes $\{\epsilon_n, n \geq 0\}$ satisfies $0 < \epsilon_n < 1$ and*

$$\sum_{n=0}^{\infty} \epsilon_n = \infty, \quad \sum_{n=0}^{\infty} \epsilon_n^2 < \infty \ ,$$

then $\lim_{n\to\infty} \gamma_n = \gamma^*$ almost surely. (ii) Suppose that the step sizes satisfy $\underline{\epsilon} \leq \epsilon_n \leq \bar{\epsilon}$ for all $n = 0, 1, \ldots,$ for some constants $0 < \underline{\epsilon} < \bar{\epsilon} < 1$. Then for any $\delta > 0$, there exists $b := b(\delta) < \infty$ such that $\limsup_{n\to\infty} \mathbf{P}[|\gamma_n - \gamma^*| \geq \delta] \leq b\bar{\epsilon}$.

The first part of the lemma states that given a proper choice of the step size sequence, the threshold γ_n converges to γ^* with probability one. Moreover, the second part of the lemma guarantees that if the step size is bounded properly, then γ_n will stay in a small neighborhood around γ^* with high probability.

BIBLIOGRAPHY

[1] B. Awerbuch, D. Holmer and H. Rubens, "High throughput route selection in multi-rate ad hoc wireless networks," in Proceedings of Wireless On-Demand Network Systems, First IFIP TC6 Working Conference, Italy, 2004.

[2] S. Bandyopadhyay, E. J. Coyle, and T. Falck, "Stochastic properties of mobility models in mobile ad hoc networks," in Proceedins of CISS, Princeton, NJ, 2006.

[3] R. Bellman, "On a routing problem," Quarterly of Applied Mathematics Vol. 16, No. 1, pp 87-90, 1958.

[4] C. Bettstetter and C. Wagner, "The spatial node distribution of the random waypoint mobility model," in Proceedings of German Workshop on Mobile Ad Hoc Networks (WMAN), Ulm (Germany), March 2002.

[5] C. Bettstetter, G. Resta and P. Santi, "The node distribution of the Random Waypoint mobility model for wireless ad hoc networks," IEEE Transactions on Mobile Computing, vol. 2, No. 3, July-September 2003.

[6] V. S. Borkar and S. P. Meyn, "The O.D.E. method for convergence of stochastic approximation and reinforcement learning," SIAM J. Control Optim., Vol. 38(2), pp. 447-469, 2000.

[7] T. Camp, J. Boleng and V. Davies, "A survey of mobility models for ad hoc network Research," Wireless Communications and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking Research, Trends and Applications, 2(5):483-502, 2002.

[8] J. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in Proceedings of IEEE Infocom, March 2000.

[9] C. Chiang, H. Wu, W. Liu and M Gerla, “Routing in clustered multihop, mobile wireless networks with fading channel,” in Proceedings of IEEE Singapore International Conference on Networks, SICON, Singapore, April 1997.

[10] D. R. Cox, *Renewal Theory*, Methuen and Co. LTD Science Paperbacks, 1970.

[11] L. R. Ford (Jr.), *Network Flow Theory*, Santa Monica, California, The RAND Corporation, 14. August, 1956.

[12] E. Gafni and D. Bertsekas, “Distributed algorithms for fenerating loop-free routes in networks with frequently changing topology,” IEEE Transactions on Communications, Vol. 29, No. 1, pp. 11-18 IEEE, January, 1981.

[13] M. Grossglauser and D. Tse “Mobility increases the capacity of ad hoc wireless networks,” IEEE/ACM Transactions on Networking, Vol. 10(4), pp. 477-486, August 2002.

[14] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” IEEE Trans. on Information Theory, Vol. 46(2), pp. 388-404, March 2000.

[15] Y. Han, R. J. La, A. M. Makowski, and S. Lee “Distribution of path durations in mobile ad-hoc networks - Palm’s theorem to the rescue,” to appear in Computer Networks (Special Issue on Network Modeling and Simulation), August 2006.

[16] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, “A group mobility model for ad hoc wireless networks,” in ACM/IEEE MSWiM, August, 1999.

[17] Z. J. Haas, “A new routing protocol for the reconfigurable wireless networks,” in Proceedings of 6th IEEE International Conference on Universal Personal Communications, IEEE ICUPC, pp. 562-566, San Diego, CA, USA, October 1997.

[18] D. P. Heyman and M. J. Sobel, *Stochastic Models in Operations Research, Volume I*, McGraw-Hill, New York, New York, 1982.

[19] A. Jardosh, E. M. BeldingRoyer, K. C. Almeroth and S. Suri, “Towards realistic mobility models for mobile ad hoc networks,” in Proceedings of MobiCom, San Diego, CA, September 2003.

[20] A. Jardosh, E. M. BeldingRoyer, K. C. Almeroth and S. Suri, "Real world environment models for mobile ad hoc networks," in the IEEE Journal on Special Areas in Communications - Special Issue on Wireless Ad hoc Networks, 2005.

[21] D. B. Johnson, and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing*, pp. 153-181, 1996.

[22] H. J. Kushner and G. George Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd ed., Springer, New York, New York, 2003.

[23] M. Kwan, K. Doganay and L. Jain "Fair multi-path selection for real-time video transmission in ad-hoc networks using artificial intelligence," *Design and application of hybrid intelligent systems* , IOS Press, Amsterdam, The Netherlands, 2003.

[24] S. Kwon and N. Shroff "Energy-efficient interference-based routing for multi-hop wireless networks", in Proceedings of IEEE Infocom, Barcelona, Spain, 2006.

[25] J.-Y. Le Boudec, "Understanding the simulation of mobility models with Palm calculus," *available at http://icapeople.epfl.ch/leboudec.*

[26] M. R. Leadbetter, "On extreme values in stationary sequences," *Z. Wahrscheinlichkeitstheorie verw. Gebiete*, Vol. 28, pp. 289-303, 1974.

[27] M. R. Leadbetter, "Extremes and local dependence in stationary sequences," *Z. Wahrscheinlichkeitstheorie verw. Gebiete*, Vol. 65, pp. 291-306, 1983.

[28] S. Lee, M. Gerla and C. Toh, "A simulation study of table-driven and on-demand routing protocols for mobile ad hoc networks," IEEE Network, Volume 13, Issue 4, pp 48 - 54, July-Aug, 1999.

[29] G.Lin, G. Noubir, and R. Rajaraman "Mobility models for ad hoc network simulation," in Proceedings of IEEE Infocom, HongKong, 2004.

[30] S. Murthy and J. J. Garcia-Luna Aceves, "A routing protocol for packet radio networks," in Proceedings of ACM International Conference on Mobile Computing and Networking, MOBICOM, Berkeley, CA, November 1995.

[31] W. Navidi and T. Camp, "Stationary distributions for the Random Waypoint mobility model," IEEE Transactions on Mobile Computing ITMMC-3, pp. 99-108, 2004.

[32] P. Papadimitratos, Z. Haas, and E. Sirer, "Path set selection in mobile ad hoc networks," in Proceedings of the Third ACM International Symposium on Mobile Ad Hoc Networking and Computing, New York, USA, 2002.

[33] V. Park and M. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in Proceedings of IEEE Infocom, April 1997.

[34] C. E. Perkins, *Ad-hoc Networking,* Addison-Wesley Longman, Incorporated, 2000.

[35] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," Computer Communications Review, Vol. 24(4), pp. 234-244, October 1994.

[36] C. Perkins and E. M. Royer, "Ad hoc on-demand distance vector routing," in Proceedings of the second annual IEEE workshop on mobile computing systems and applications, pp. 90-100, February 1999.

[37] S. Ramanathan and E. Lloyd, "Scheduling algorithms for multi-hop radio networks," in IEEE/ACM Transactions on Networking, Vol. 1(2), pp. 166-177, April 1993.

[38] H. Robbins and S. Monro, "A stochastic approximation method," Annals of Mathematical Statistics, vol. 22, pp. 400– 407, 1951.

[39] V. Rodoplu and T. Meng "Minimum energy mobile wireless networks," IEEE Journal on Selected Areas in Communications, Vol. 17(8), pp. 1333-1344, August 1999.

[40] N. Sadagopan, F. Bai, B. Krishnamachari, and A. Helmy, "PATHS: Analysis of path duration statistics and the impact on reactive MANET routing protocols," in Proceedings of ACM MobiHoc, Annapolis, MD, June 2003.

[41] J. Tang, G. Xue and W. Zhang, "Interference-aware topology control and QoS routing in multi-channel wireless mesh networks," Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing,Urbana-Champaign, IL, USA, 2005.

[42] J. Tang, G. Xue, C. Chandler, and W. Zhang, "Interference-aware routing in multihop wireless networks using directional antennas," in Proceedings of IEEE Infocom, 2005.

[43] C.-K. Toh, *Ad-hoc mobile wireless networks: protocols and systems*, Prentice Hall PTR, 2001.

[44] G. S. Watson, "Extreme values in samples from m-dependent stationary processes," Ann. Math. Statist. Vol. 25, pp. 798-800, 1954.

[45] W. Wei and A. Zakhor, "Path Selection For Multi-path Streaming in Wireless Ad Hoc Networks," to appear in proceedings of International Conference on Image Processing, Atlanta, Georgia, September, 2006.

[46] H. Zhang, "A Numerical study of link and path durations in mobile ad hoc networks," Master thesis, 2006, available upon request.

[47] R. D. Yates and D. J. Goodman, *Probability and stochastic processes*, Wiley, 1999.

[48] J. Yoon, M. Liu and B. Noble, "Random Waypoint Considered Harmful," in Proceedings of IEEE Infocom, vol 2, pp 1312-1321, San Francisco, CA, April 2003

[49] http://wiki.uni.lu/secan-lab/Dynamic+Source+Routing.html

[50] http://wiki.uni.lu/secan-lab/Ad+Hoc+On-Demand+Distance+Vector+Routing+Protocol.html

[51] http://www.isi.edu/nsnam/ns/