Communication Networks
University of Bremen
Prof. Dr. rer. nat. habil. C. Görg

Dissertation

Wireless Multi-hop Ad hoc Networks: Evaluation of Radio Disjoint Multipath Routing

of
Koojana Kuladinithi

from

Galle, Sri Lanka

22nd of December 2009

First Examiner:	Prof. Dr. rer. nat. habil. C. Görg
Second Examiner:	Prof. Dr. -Ing. Andreas Timm-Giel
Submitted on:	22nd of December 2009

ACKNOWLEDGEMENT

This thesis was written during my research assistantship at the Communication Networks Group (ComNets) of the Center for Computer Science and Information Technology (TZI) at the University of Bremen. I sincerely thank the many people who have contributed in many different ways to make this work possible.

Working for ComNets has been a very enriching experience since I got the opportunity to work side by side with some great people. Prof. Ranjit Perera introduced me to ComNets, such a nice group of people whom I consider as my extended family.

Prof. Carmelita Görg made all the arrangements to commence my assistantship here. She guided me and gave me invaluable advice that provided me the direction required for my research work. She was able to place me in the right projects that let me undertake research related to my thesis area. She has been a role model for me and there are 2 things that I would like most to emulate from her. The first is the discipline she has in her professional life and her social life that I know. The second is her efforts that she makes to provide opportunities for women in academia.

Prof. Andreas Timm-Giel is a person I always relied on to give me advice and ideas. The innumerable discussions I had with him on topics ranging from project work to my PhD have resulted in bringing clarity to my research.

Prof. Samir Das had discussions with me that motivated and gave me some critical ideas that became part of my PhD focus. The early discussions that I had with Ioannis Fikouras and Niko Fikouras as part of the NOMAD project gave me the initial direction in my research work. I got many valuable ideas and help when performing some of the wireless related experiments of the wearIT@Work project from Philipp Hofmann and Christian Bettstetter.

I will never forget the support and the beneficial discussions I had with my current and former colleagues, especially the OPNET experts (Thushara Weerawardane, Xi Li, Yasir Naseer Zaki), Mathlab expert Liang Zhao and networking experts (Chunlei An, Andreas Könsgen, Amanpreet Singh, Bernd-Ludwig Wenning and Markus Becker). It was a pleasure to have worked in different projects with my colleagues. Martina Kammann's administrative skills and Karl-Heinz Volk's technical skills were always available whenever I needed them. The students whom I have supervised, especially Chunlei An, Xiao Sun and Stephane Batassi, have provided me with insights into related research areas through their work on Diploma and Master Theses.

All this work would have been practically impossible if it were not for Hasini, my little daughter, being happy at the Uni-Kids daycare. She enjoyed her stay at Uni-Kids that provided me the opportunity to work peacefully. Last but not least, Asanga, my husband supported me not only concerning personal matters but also with technical issues. I can say that I had round-the-clock tech support, especially related to programming issues.

Finally, it is with reverence and respect that I remember my mother and my late father for creating an environment for me to pursue my education with all the hardships.

ABSTRACT

A wireless multi-hop ad hoc network consists of a collection of nodes, which can communicate without any fixed base stations or networking infrastructure. Multi-hop ad hoc networks are ideally suited in areas such as sensor networking, community networking and networking used in emergency situations. Since transmission is wireless and nodes could be mobile, ad hoc networks bring about new challenges to be considered when designing routing algorithms.

Multipath routing discovers more than one route between a source node and a destination node in a wireless multi-hop ad hoc network. These routes can be used simultaneously to distribute traffic among several paths or used as backup paths. Multipath routing can provide benefits such as load balancing, bandwidth aggregation, fault tolerance and improvement in QoS. The work done in this thesis investigates the simultaneous use of multipath routes in wireless multi-hop ad hoc networks.

In wireless multi-hop ad hoc networks, the simultaneous use of multiple routes may degrade the performance of applications due to mutual interference of discovered paths, irrespective of whether paths are physically node disjoint or link disjoint. Therefore, the selection of non-interfering routes is the main criterion to be addressed when using multiple routes simultaneously. This thesis introduces a new metric to select multiple routes by reducing the effect of interference between paths as far as possible and also selecting the least congested paths. The proposed protocol is named Radio Disjoint Multipath (RDM). The concept of the RDM protocol which can be applied to both reactive and proactive ad hoc protocols is developed and feasibility of the protocol is proven by an implementation and also through an analytical model.

Furthermore, this thesis introduces a novel mechanism to distribute multiple flows as well as packets of a single flow based on the properties of the discovered path, which is computed considering the Background Traffic Load (BTL) of each path and the mutual interference between paths. The single flow distribution is further investigated by replicating packets among the RDM paths. This distribution is used to enhance the reliability in adverse environments such as a fire-fighting scenario.

The evaluation of results is done considering the non-interfering RDM routing, the interfering RDM routing and the single path routing. When using the RDM routes, two distribution methods, viz., the single flow and the multiple flow distribution methods are considered. The performance of the applications is compared using real application flows consisting of audio conferencing, video transmissions, HTTP web accessing and FTP downloads that use different scenarios with and without mobility. The analysis shows that the use of non-interfering RDM routes simultaneously to distribute application flows significantly outperforms the use of single path routing for most of the scenarios investigated.

In summary, all investigations presented in this thesis can help to enhance the application performance in different kinds of wireless multi-hop ad hoc networks of Mobile Ad hoc NETworks (MANET), Wireless Sensor Networks (WSN) and Wireless Mesh Networks (WMN), by discovering RDM routes and using them simultaneously.

KURZFASSUNG

Ein drahtloses Multi-Hop-Ad-Hoc-Netz besteht aus einer Menge von Knoten, die ohne feste Basisstationen oder Netzinfrastruktur miteinander kommunizieren können. Multi-Hop-Ad-Hoc-Netze sind für Anwendungsfälle wie Sensornetze, freie Funknetze und Netze für Notfallsituationen sehr gut geeignet. Da die Übertragung drahtlos ist und die Knoten beweglich sein können, führen Ad-Hoc-Netze zu neuen Herausforderungen, die beim Entwurf von Routingalgorithmen beachtet werden müssen.

Mehrwege-Routingverfahren ermitteln mehrere Routen zwischen einer Quelle und einer Senke. Diese Routen können gleichzeitig verwendet werden, entweder um den Verkehr auf verschiedene Pfade aufzuteilen oder als Ersatz-Pfade. Mehrwege-Routing bietet Vorteile wie Lastverteilung, Bandbreiten-Aggregation, Fehlertoleranz und Verbesserung der Dienstgüte. Die im Rahmen dieser Arbeit durchgeführten Untersuchungen betreffen die gleichzeitige Nutzung von Mehrwege-Routen in drahtlosen Multi-Hop-Ad-Hoc-Netzen.

In drahtlosen Multi-Hop-Ad-Hoc-Netzen kann die gleichzeitige Verwendung mehrerer Routen die Leistung von Anwendungen aufgrund gegenseitiger Störung beeinträchtigen, unabhängig davon, ob die Pfade in Bezug auf ihre Knoten oder Verbindungen physikalisch disjunkt sind. Daher ist die Auswahl interferenzfreier Routen das Hauptkriterium, das berücksichtigt werden muss, wenn mehrere Routen gleichzeitig verwendet werden. Diese Arbeit stellt eine neue Metrik zur Auswahl mehrerer Routen vor, bei der die Auswirkung von Interferenz zwischen Pfaden so weit wie möglich reduziert wird und darüber hinaus die am wenigsten ausgelasteten Pfade gewählt werden. Das vorgeschlagene Protokoll wird als Radio Disjoint Multipath (RDM) bezeichnet. Das Konzept dieses RDM-Protokolls, das sowohl auf reaktive als auch auf proaktive Ad-Hoc-Protokolle angewendet werden kann, wird im Rahmen dieser Arbeit entwickelt, und die Durchführbarkeit des Protokolls wird durch eine Implementierung und durch ein analytisches Modell nachgewiesen.

Darüber hinaus stellt diese Arbeit ein neues Verfahren vor, das sowohl mehrere Flüsse als auch Pakete eines einzelnen Flusses basierend auf den Eigenschaften des ermittelten Pfades verteilt. Die Berechnung dieser Eigenschaften berücksichtigt die Hintergrundlast (Background Traffic Load, BTL) jedes Pfades und die gegenseitige Interferenz zwischen Pfaden. Die Verteilung eines einzelnen Flusses mittels Replikation von Paketen auf den RDM-Pfaden wird weiter untersucht. Diese Verteilung wird verwendet, um die Zuverlässigkeit in ungünstigen Umgebungen, beispielsweise in einem Brandbekämpfungsszenario, zu verbessern.

Die Auswertung der Ergebnisse erfolgt unter Berücksichtigung des interferenzfreien RDM-Routings, des interferenzbehafteten RDM-Routings und des Einzelpfad-Routings. Bei der Verwendung des RDM-Routings werden zwei Verteilungsmethoden, nämlich die Verteilung von Einzelflüssen und die Verteilung von mehreren Flüssen, berücksichtigt. Die Leistungsfähigkeit der Anwendungen wird unter Verwendung realer Anwendungsflüsse, bestehend aus Audiokonferenzen, Videoübertragungen, HTTP-Netzzugriffen und FTP-Downloads, verglichen, die verschiedene Szenarien mit und ohne Mobilität verwenden. Die Analyse zeigt, dass die Verwendung von interferenzfreien RDM-Routen zur Verteilung von Anwendungsflüssen in den meisten untersuchten Szenarien deutlich leistungsfähiger ist als die Verwendung eines Einzelpfad-Routings.

Zusammenfassend können alle Untersuchungen, die in dieser Arbeit präsentiert werden, dazu beitragen, die Leistungsfähigkeit verschiedener Arten von drahtlosen Multi-Hop-Ad-Hoc-Netzen wie mobilen Ad-Hoc-Netzen, drahtlosen Sensornetzen und drahtlosen Mesh-Netzen zu verbessern, indem RDM-Routen ermittelt und gleichzeitig verwendet werden.

TABLE OF CONTENTS

1. Introduction

Wireless ad hoc communication is suitable for environments where there is no possibility of using wired or infrastructure based communications. The multi-hop ad hoc networks of Mobile Ad hoc NETworks (MANET), Wireless Sensor Networks (WSN) and Wireless Mesh Networks (WMN) have drawn extensive attention in recent years. However, the performance of wireless multi-hop ad hoc networks in different environments is still open for research. This chapter gives an overview of the current state of the art of wireless multi-hop ad hoc networks, followed by the research goals that are achieved to improve multi-hop ad hoc networking within the scope of this thesis. The last section gives an outlook to each subsequent chapter.

1.1 Overview of Wireless Multi-hop Ad hoc Networks

The nodes that are not within the direct communication range in wireless ad hoc multi-hop networks, use the other nodes as a relay to forward the data packets to a target destination. A MANET [1] is a collection of nodes that does not connect to a fixed base station or other existing networking infrastructure. MANETs can be deployed in military environments and emergency situations such as fire-fighting coordination and earthquakes [2-4]. A WSN also has similar characteristics to MANET, but the number of nodes in WSNs is usually much larger than in a MANET. Furthermore, sensor nodes have very limited battery power, computational capabilities and storage. A WMN is also a multi-hop network which can be configured to work in ad hoc or infrastructure based modes. Compared to MANETs and WSNs, the nodes in a WMN are always stationary and most likely to be mains powered.

Conventional IP based routing protocols are not appropriate for these networks because of the temporary nature of the network links and additional constraints on nodes i.e. limited bandwidth and power [5]. Routing protocols for such environments must be able to keep up with the high degree of node mobility or link failures that often change the network topology unpredictably, specifically in MANETs and WSNs. Therefore, the following issues have to be considered, when developing a routing protocol for ad hoc multi-hop networks [6].

- *Dynamic topology*: nodes can move in and out of the transmission range of neighboring nodes at any time depending on the type of movement. Furthermore, nodes can stop functioning due to the expiry of battery lifetime, physical damage, or heavy congestion at the node (e.g. packets dropped due to buffer overflows). The above changes in a network result in frequent topological changes in the routing. Therefore ad hoc routing should be able to detect the availability of the connectivity and re-discover routes in case of active link failures. The reaction in the routing protocol has to be fast enough and

transparent to the upper layers so that performance of the applications is not interrupted [7].

- *Limited network bandwidth:* Most of the wireless technologies deployed in ad hoc networks have limited bandwidth. When sharing the same channel among multi-hop ad hoc nodes, the usable bandwidth per node decreases further. Therefore, ad hoc routing protocols have to be designed to work efficiently in bandwidth limited networks.
- *Quality of the wireless link*: From the reliability perspective, wireless links are error-prone. The data transmission can be affected due to many reasons such as contention, interference from other communications, varying environmental conditions, etc. Therefore, ad hoc routing protocols should be able to discover the best routing path by evaluating the available paths dynamically [8].

1.1.1 Types of Ad Hoc Routing Protocols

There are 3 main categories of ad hoc routing protocols available as shown in Figure 1-1. They are flat routing, hierarchical routing and geographic routing. In flat routing, each node plays an equal role when forwarding data. The MANET working group [9] at the IETF mainly focuses on standardizing the flat routing protocols, which are further classified into two classes:

- *Reactive (source-initiated & demand driven):* routes to the destination are found on-demand (when the source wants to send data).
- *Proactive (table-driven):* Routes are found and maintained for all nodes in the network, irrespective of their actual use.

Proactive MANET protocols of OLSR (Optimized Link State Routing) [10] and TBRPF (Topology Based Reverse Path Forwarding) [11] and reactive protocols of AODV [12] and DSR (Dynamic Source Routing) [13] are published as experimental RFCs, as of this writing. The DYMO (Dynamic Manet On-demand) [14] protocol which is designed based on both AODV and DSR protocols, has been agreed to be published as a standard RFC.

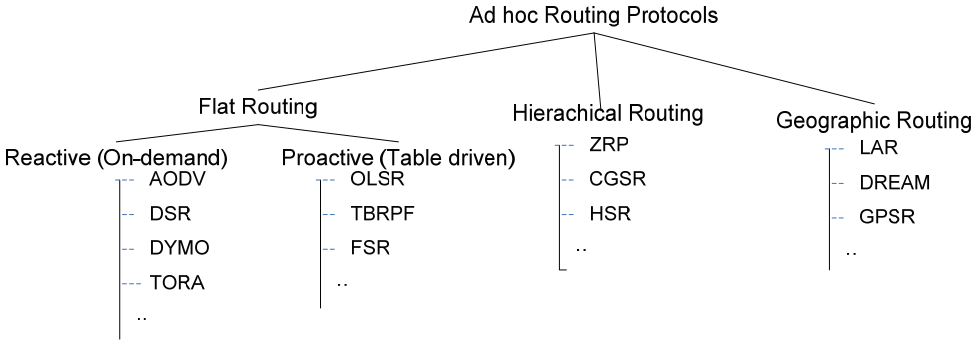


Figure 1-1 Types of ad hoc routing protocols

In contrast to flat routing, hierarchical routing usually assigns different roles to the nodes. When the size of the network grows, flat routing is not feasible due to the

processing overhead. Therefore, hierarchical routing is used with a hierarchical addressing system. Geographic routing requires each node to be equipped with a location finder system such as Global Positioning System (GPS). The location information can be used for directional routing in distributed ad hoc networks. For example, LAR (Location Aided Routing) performs the route discovery through limited flooding using location information.

1.1.2 Multipath Ad hoc Routing

Most of the protocols in Figure 1-1 are designed to discover a single routing path. Multipath routing allows the use of multiple routing paths between a given source-destination pair. Multipath routing always has been a favorable alternative for today's communication networks, as it can be used to distribute traffic in the network and also as a fault tolerant mechanism. In multi-hop ad hoc networks, the routing environment changes rapidly. Therefore multipath routing is an alternative to improve the application performance in ad hoc multi-hop environments. Multiple routing paths can be used simultaneously to distribute traffic among several paths or use one path at a time to reduce route re-discoveries. There are numerous multipath routing protocols proposed for ad hoc networks as discussed in Chapter 2. Performance benefits that can be achieved for wireless multi-hop ad hoc networks with multipath routing are:

- *Load balancing:* Multipath routes can be used to distribute different applications to ease the burden of the congested or over-utilized wireless links. Therefore, the application performance can be improved by reducing end-to-end delays and buffer overflows.
- *Aggregation of bandwidth:* By splitting packets of a single application to the same destination among multiple paths, the effective bandwidth can be aggregated. When there are multiple paths with very low bandwidth compared to the bandwidth required by the application, splitting can be used to improve the application performance by using multiple paths.
- *Fault tolerance/Increased reliability:* Multipath routing protocols can improve the reliability by replicating packets to the destination via alternative paths. The replication increases the reliability in lossy environments such as a fire-fighting scenario, where reliability of the data communication is the most important aspect [2].
- *Less route discoveries:* In Single Path (SP) routing, a route failure means that a new path discovery needs to be initiated. This results in a route discovery delay. The delay is minimized in multipath routing by redirecting applications to another active path without initiating a re-discovery of a new path.
- *QoS aware routing:* Discovery of multiple routing paths with different characteristics can be used to distribute application based on different criteria [15]. For example, real time traffic can be forwarded via least congested paths, while bursty traffic can be directed to the paths with higher delays.

1.2 Motivation and Research Goals

Previous research in wireless multi-hop ad hoc networks mostly focuses on the utilization of multiple paths as backup paths in case of failures in the primary routing path (see Table 2-1). Multipath routing can also be used to improve communication efficiency and promote quality of service by utilizing more than one path simultaneously. It helps to improve the application performance due to distribution of traffic between paths, increased reliability, and the optimal utilization of otherwise unused paths. Multipath routes can be utilized simultaneously for:

- *Multiple flow distribution:* Distribution of independent flows among multiple routing paths. A flow is a sequence of packets that are originated by an end point to another end point, identified by a tuplet of information: source address, destination address, transport protocol (UDP, TCP), source port and destination port. Better application performance and distributed load is seen as key advantages gained with multiple flows distributed to different paths.
- *Single flow distribution:* This can be used in two ways. Firstly, by splitting packets of a flow among multiple routing paths to balance the load in the network. This improves the performance by reducing the congestion. Secondly, replicating packets of a flow to all routing paths to increase the reliability in lossy environments such as a fire-fighting scenario. In such scenarios, replicating packets of a flow in spite of the degradation of overall network performance and throughput is acceptable.

However, simultaneous use of multiple routing paths can result in reduced performance in multi-hop ad hoc networks due to the fact that all nodes are using the same radio channel. For example, in an IEEE 802.11x based multi-hop ad hoc network, all nodes within the interference range are competing for the same channel using the well-known CSMA/CA schemes.

There are mainly three methods that can be used to avoid the interference in multi-hop ad hoc networks. They are radio resource management (e.g. use of directional transmission, use of two different channels, etc.), scheduling of the transmission (e.g. STDMA) and avoiding interfering paths at the routing layer. This work focuses on enhancing the routing layer to avoid the use of interfering paths.

This thesis introduces a new metric to select multiple routes by reducing the effect of interference between nodes as far as possible and also to select the least congested paths, which is termed Radio Disjoint Multipath (RDM). The use of node disjoint paths with a minimum or no radio interference helps to avoid performance degradation (e.g. the flow in the middle problem in WLAN) [16]. This thesis focuses on the investigation of the following research areas when developing the RDM routes to be used in wireless multi-hop ad hoc networks.

- *Discovery of interference aware routes:* This thesis introduces a mechanism to discover interference aware multiple routes, which can be implemented at the network layer. The network layer of the protocol stack has been chosen to

develop the RDM protocol due to two major reasons. Using a lower layer means that every bearer technology (IEEE 802.11a, b, g, Bluetooth, etc.) requires specific mechanisms suitable for each technology. Secondly, to go up the protocol stack (Transport/Application) means that many applications have to be modified to cater for the behavior required by ad hoc networking. Therefore, the RDM protocol can be used without any modifications to upper layers (e.g. TCP/UDP) or lower layers (e.g. IEEE WLAN) protocols. Most previous research on multipath routing proposes the simultaneous use of paths by modifying IEEE 802.11x protocols or the upper layer protocols such as TCP (see section 4.1.3). The discovery of RDM routes are done considering two criteria viz., the mutual interference between paths and the existing Background Traffic Load (BTL) of a path. Though this thesis does not focus on the discovery of paths via multiple interfaces with different wireless technologies, the concepts proposed in this work can be used for these kinds of wireless multi-hop ad hoc networks as well. Further, an analytical model is introduced to discover the RDM routes by modeling the interference between paths together with the BTL.

- *Distribution methods:* This thesis introduces 3 different distribution methods called multiple flow distribution, splitting of a single flow and replication of a single flow. It further proposes the distribution algorithms based on the bandwidth measured considering the interference between paths and the existing BTL (Background Traffic Load).
- *Evaluation of results:* A detailed analysis of results is done by implementing the RDM protocol in the OPNET simulator. The results are taken using stationary and mobile topologies. The evaluation of results is done considering SP and simultaneous use of interfering and non-interfering routes.

1.3 Document Structure

This thesis consists of the following chapters.

- Chapter 2 is a review of existing multipath ad hoc routing protocols. It compares the mechanisms used in different proposals highlighting the performance comparison of simultaneous use of interfering and non-interfering routes.
- Chapter 3 is devoted to explaining the detailed operations of the RDM protocol. It also gives an outlook to the implementation of the RDM protocol in the OPNET simulator and discusses how to implement the RDM protocol in real environments.
- Chapter 4 discusses the performance issues when using TCP and UDP based applications in wireless multi-hop ad hoc networks. The explanations given in this chapter are used to justify the results that are discussed in chapter 5.
- Chapter 5 analyzes the performance of the RDM protocol in different scenarios. The results are taken with multiple flow distribution and splitting of a single flow.
- Chapter 6 details the performance of replicating packets among RDM routes focusing on a fire-fighting scenario.

- Chapter 7 introduces an analytical model developed in this thesis to model the discovery of RDM routes.
- Chapter 8 concludes the results of this thesis and gives an outlook to further work.
- Chapter 9 and 10 provide appendices that include further elaborations of topics discussed in the main text of this thesis.

2. Related Work: A Review of Research on Multipath Routing

Multipath routing creates multiple paths between a pair of source and destination nodes for a given communication. Multipath routing is an improvement to single path routing to provide backup paths in case of path failures to prevent further route discoveries and also to distribute flows (i.e. application data) to increase the effective bandwidth. The first section explains the basics of single path routing. The next section discusses concepts and techniques used in developing multipath routing protocols for wireless multi-hop ad hoc networks. The third section reviews the multipath routing protocols used in previous research. The last section concludes the chapter by giving an overview to Radio Disjoint Multipath (RDM) based routing introduced in this thesis, comparing its features with previous work.

2.1 Single Path Routing

The on-demand routing approach is mostly used for Single Path (SP) routing in wireless multi-hop ad hoc networks as it has a lower overhead and reacts faster to mobility [1] [17, 18]. Most of the multipath research is extended based on SP on-demand ad hoc protocols. On the other hand, extending SP to multipath is not challenging for proactive protocols, since each node knows about the available routes in the network beforehand. The use of multipath routing for on-demand protocols results in reducing the number of route discoveries and the discovery of more routing paths dynamically with differing properties. Therefore, multipath routing based on on-demand protocols can be used to select paths dynamically and to distribute applications based on different criteria. This section explains the basic operations of on-demand SP routing before detailing the multipath routing.

Instead of periodically exchanging route messages to maintain a permanent route table of the full topology, on-demand routing protocols build routes only when a node needs to send data packets to a destination. The source floods the network to search for the destination and discover the route. Once routes are made, they are held only while data utilizes these routes. Link failures during communications are detected by the unaffected nodes in a path. Since no periodic route table exchange is required, control overhead is minimized and the routing information is utilized efficiently. On-demand ad hoc protocols have a minimum of 4 control messages to organize routing paths for communications. Figure 2-1 shows that the originator, denoted by S, is in need of a path to communicate with a destination, denoted by D. The coverage area of each device is

represented by the circles. I1, I2 and I3 denote the intermediate nodes in the path to reach D.

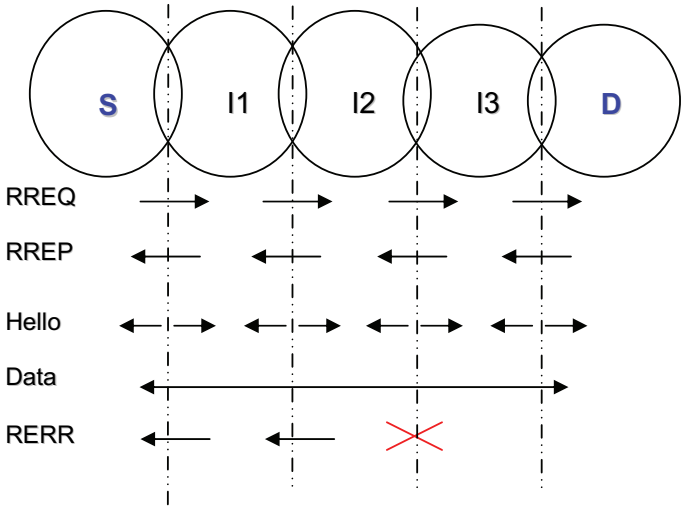


Figure 2-1 Control messages used in on-demand SP protocol: route discovery using RREQ, route establishment with RREP, route maintenance using Hello and the detection of link failure using RERR

Since S is unaware of the location of D, it broadcasts the Route Request (RREQ) message asking where D is located. This message is forwarded by each intermediate node until it reaches D. During this process, called the route discovery, each node creates the routing path to S. This path is called the reverse path. Once D receives RREQ, it sends a Route Reply (RREP) message to S over the previously set routing path (reverse path) towards S. All nodes update the routing paths towards D during the propagation of the RREP message. This is called the forward path. Once S receives the RREP, it can start transmitting the data. During the discovery of the path, S can buffer the data to release them when the path is made. In order to maintain the discovered path, each node has to detect link breakages to neighboring nodes. This can be done by monitoring data transmission on links to neighbors. The connectivity to immediate neighboring hops can be detected by using either link layer acknowledgements or network layer acknowledgements (e.g. Hello messages in AODV). Once a node detects that an immediate hop is not reachable (e.g., I2 has ceased to hear Hello messages from I3), it should inform the other immediate hop, using Route Error (RERR) messages. This message can be propagated all the way to S or any one of the other intermediate nodes, to utilize a different path. Utilization of a different path may mean a new route discovery or the use of another known path to D [5].

During the propagation of RREQ messages, the creation of routing loops has to be avoided. In AODV [1], this is done by intermediate nodes by forwarding the RREQ only once for a route discovery. A route discovery is uniquely identified by the addresses of S and D together with a sequence number generated by S for each new route discovery. In DSR [13], routing loops can be detected immediately as RREQ messages carry the constituent node identifications of the route in the message itself.

2.2 Multipath Routing

When developing a multipath routing protocol, at least three main criteria have to be considered:

1. Multipath route discovery: The route discovery process defines one or more criteria to select multiple routes, which can be processed at the destination, intermediate nodes or the source.
2. Utilization of multiple routes: The utilization of paths defines how to distribute traffic flows among multiple paths.
3. Path maintenance, path evaluation and re-discovery: The path maintenance process defines how to maintain multiple paths that are already discovered. The path evaluation process makes decisions about when to change the paths by evaluating the quality of the discovered paths. The re-discovery of routes has to be initiated upon failure of all available paths.

2.2.1 Multipath Route Discovery Process

There are several criteria to be used in selecting multiple paths for a given source-destination pair. The most commonly used criterion is to use disjoint paths. In principle, disjoint paths offer more aggregate resources and higher fault-tolerance than non disjoint paths. Since non disjoint paths share links and nodes, a link or a node failure may affect all the paths. There exist three types of disjoint paths as shown in Figure 2-2.

- Node Disjoint Paths: no common nodes between paths except for the source and the destination. This guarantees that links fail independently and can be used for load balancing purposes.
- Link Disjoint Paths: no common links between paths, but there can be common intermediate nodes. The multiple links which go through a common node might fail together if the common node moves out of the range.
- Partial Disjoint Paths: Selected paths might have common nodes or links. When a link in the primary path fails, the other available alternative links can be used as an alternative path. The alternative path may share the rest of active nodes or links in the primary path.

Disjoint paths based on above criteria may not be the optimum solution to discover better paths, if paths consist of too many hops [19]. As packets may get diverted towards a longer path unnecessarily, a higher number of hops increases the end to end delay and wastes more bandwidth [20] [21]. Therefore most of the selection algorithms consider the hop count of a path as well. In addition to hop counts, criteria such as delay, available bandwidth, Signal to Noise Ratio (SNR), etc. can also be taken into account when selecting better paths [22] [23] [19].

When discovering multiple routes, the intermediate node should not suppress the forwarding of duplicated RREQ messages in contrast to SP routing. Otherwise multipath route discovery also tends to find only a SP. For example, if node 3 in Figure

2-2–(b) does not forward a later RREQ that it receives from the node 8, the path via node 7 and node 8 will not be discovered in this topology. On the other hand, if the node 3 allows the forwarding of all RREQ messages, there might be routing loops. For example, the reverse paths via "S, 1, 7, 1, 2" create routing loops.

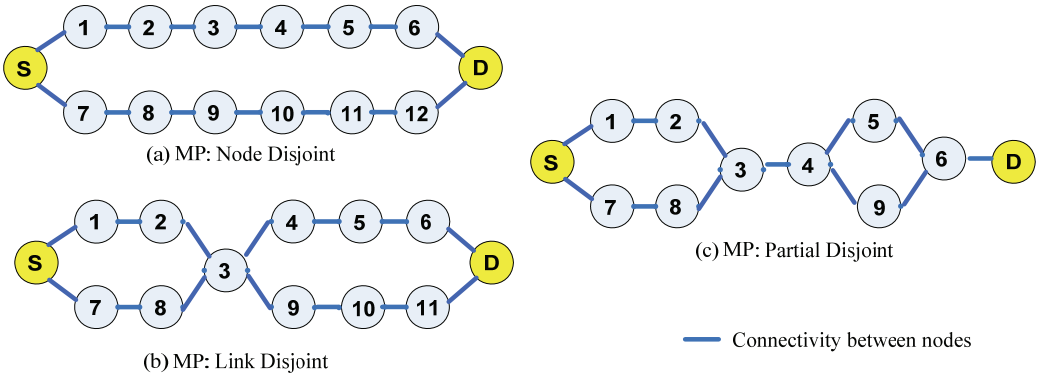


Figure 2-2 Types of multipath routing: Node Disjoint, Link Disjoint and Partial Disjoint

Therefore, SP route discovery process has to be modified in multipath routing to increase the route diversity by avoiding routing loops and also reducing the number of RREQs flooded in the network. This requires intermediate nodes and the destination node to create more than one route towards the source and vice versa. Some methods that are used in existing work are explained below in brief.

- The straight forward way to check for loops is by every node maintaining complete path information. This solution can be implemented easily for protocols like DSR and DYMO. The DSR protocol [13] has a unique advantage by using source routing. As the route is part of the packet itself, routing loops cannot be formed as they can be immediately detected. This is also applicable for the DYMO protocol [24] as it carries all intermediate nodes' addresses in the RREQ/RREP message. Since the source and the destination get to know all the details of each path, they can easily select node disjoint or link disjoint routes among multiple paths discovered. But, this solution introduces a higher overhead by carrying the node details in RREQ/RREP messages and also to check for loops at each intermediate node by comparing the path details.
- Since there are more overheads when forwarding all RREQs received, the intermediate node can forward only one RREQ, but keep all the reverse paths in a temporary cache. The destination is allowed to send multiple RREPs received via disjoint next hops. When forwarding the RREP back to the source, the intermediate nodes can decide which reverse paths to select to reach the source [25]. For example, node 3 in Figure 2-2–(b) can suppress the later RREQ receipts from node 8, but keeps a route to S via node 8 in its temporary reverse route cache. Overhearing of the propagation of RREPs on the other paths helps each intermediate node to decide to not participate in creating the common forward path. For example, node 12 in Figure 2-2-(a) has 2 possibilities to send RREPs to the source, i.e. via node 5 or node 11. Assuming that node 5 has already forwarded a RREP to the source and this is overheard by node 12,

therefore, node 12 should not forward later RREPs that come from the destination to node 5. But, it can forward a RREP to node 11. In this manner, only node disjoint routes are discovered by avoiding the routing loops.

- In general, routing loops can be avoided by not forwarding a RREQ which has a higher hop count to the source. Forwarding the restricted number of RREQs by considering different criteria such as hop counts, delay [26] is also used to avoid creating route loops. If a RREP includes of the last hop of a path, which is the node immediately preceding the destination on a path, each intermediate node can identify the link disjoint path. This is done by forwarding RREPs, which are uniquely identified from the previous hop and last hop pair [26]. The previous hop refers to the node which forwards the RREP message. In Figure 2-2–(b), node 3 forwards 2 RREPs generated by D since they have unique pairs of previous hop and last hop; i.e. "node 4 & node 6" and "node 9 & node 11". There exist 2 link disjoint paths that can be created in this manner. But for the topology in Figure 2-2–(c), node 3 does not forward 2 RREPs coming from the destination, since they have a common pair of previous hop and last hop; i.e (node 4 – node 6). In this case, it forwards only the first RREP it receives either via an upper or a lower path.

2.2.2 Utilization of Multiple Routes

Multiple paths can be used to improve the network performance by utilizing one path at a time (called alternate path routing or backup path routing) or utilizing two or more paths simultaneously.

2.2.2.1 Alternate Path Routing (APR)

APR uses a single route at a time to send all active flows. It uses the best available path as its primary path to send flows first. The next available best path is used upon the failure of the primary path. The next route discovery process starts after utilizing all available paths. In this situation, path maintenance has to be implemented additionally for the paths which are not used to send data. Most of the protocols based on APR select partial disjoint paths. Network performance can be improved with the APR by avoiding unnecessary route discoveries.

2.2.2.2 Simultaneous Use of Multipath (SUM) routing

SUM routing uses a selected number of discovered paths simultaneously to distribute flows. However, the drawback of using multiple paths simultaneously to distribute flows in wireless multi-hop ad hoc networks is that the nodes located in the vicinity in the other paths, may interfere by their own communications. This may result in drastic reductions in the effective bandwidth/throughput. This phenomenon is most common in multi-hop ad hoc wireless environments, when sharing the media with other nodes to prevent packet collisions. If the mutual interference between paths is very low, the SUM routes improve the application performance by aggregating the available bandwidth and

hence reducing the end to end delays and lost packets (see section 2.4.1). SUM routing requires addressing issues such as how different flows are identified and distributed, the amount of traffic that has to be put in each path, packet reordering at the destination, etc. Multiple routes can be utilized simultaneously by distributing independent flows [16], by splitting (distributing) packets of a single flow [27] or by replicating packets of a flow among all active paths. Splitting packets of a single flow can be used either when a single flow is present or when multiple flows are present. Replication increases the reliability in lossy environments such as a fire-fighting scenario, where reliability of the data communication is the most important aspect [2].

There are a few distribution algorithms that are used when distributing independent flows or splitting packets of a single flow. The individual flow can be identified based on different types of criteria such as types of protocols, port numbers, etc. [28]. The prevalent algorithm to split packets of a single flow is based on the round robin distribution. The number of packets that are sent once can be determined by different criteria such as the congestion on the paths, RTT of each path, etc. [29-31]. There might be out-of-order packet delivery, when paths do not have similar characteristics. The out-of-order packet delivery can be misinterpreted by TCP as network congestion and that will result in a poor performance. When replicating packets, removal of the redundant copy at the receiver has to be taken into account since the TCP receiver sends an acknowledgement to each copy of the packet and the TCP sender reacts negatively for the receipt of consecutive replicated acknowledgements [32].

2.2.3 Path Maintenance, Path Evaluation and Re-discovery

In wireless networks, routes may fail due to node failures (e.g. lack of battery power) or node mobility. In multipath routing, the path maintenance is done in a similar way as for the SP routing. Each node can detect the availability of active neighbors with the receipt of a data packet, a Hello message or a link layer acknowledgement. However it is difficult to detect the connectivity to neighbors in this manner, when using multipath routes for APR. The nodes in the alternate paths are not used to transmit data until the primary path fails. Therefore, these nodes have to maintain the connectivity with the neighbors in the absence of data transmission. This can be done by overhearing the data transmission on the primary path if the nodes in the alternate path are within the transmission range of the other path. Otherwise, the alternate paths have to be maintained proactively sending additional control messages. Path maintenance is simpler when using multipath for SUM as data is transmitted over all the paths simultaneously.

Once a link failure is detected, the intermediate node can send a RERR message to the source to start route discovery again or can make a decision to change to another active path that is already discovered. The source can initiate the re-discovery of routes after each path failure, or when all the paths have failed.

Even before detecting the link breakages, algorithms can be implemented to measure the quality of links and then decide to change the paths if the current QoS parameters of

a path go below an accepted range [23] [19]. This is called the path evaluation process. This requires dynamic algorithms to constantly monitor the quality of the links.

2.3 Proposals for Multipath Routing Protocols

There are several research studies that focus on discovering multiple routing paths in wireless multi-hop ad hoc networks. Most of the multipath protocols are developed by extending the on-demand MANET protocols. This section gives an overview of the existing multipath protocols.

2.3.1 AODV-BR: Backup Routing for AODV

The route discovery of AODV-BR [33] is similar to the propagation of RREQs in standard AODV, while the processing of RREP is modified to create the alternate paths. Once an intermediate node overhears a RREP packet transmitted by its neighbor on the primary path, each intermediate node records that neighbor as the next hop to the destination in its alternate routing table. Therefore, AODV-BR can establish a primary path and alternate paths during the propagation of a RREP. Data packets are delivered along the primary path. Once an intermediate node detects a link breakage on the primary path, it broadcasts the data packet only to its neighbors. This requires a modification to the header of a data packet to inform that it searches for an alternate path. If neighbor nodes have an entry for the destination, it can forward the packet by unicasting it to a next hop node of the alternate route table. This node also has to check whether the packet has been received by the next hop properly. The node that detected the link break sends the RERR to the source to initiate a route re-discovery to find an optimal route for the current topology while data are transmitted via the alternate paths. How to handle the receipt of more than one copy of a data packet at the destination is not discussed. The route life time of the alternate paths are updated by overhearing the data transmissions.

Results show that AODV-BR performs better than standard AODV due to a lower number of route discoveries. AODV-BR attempts to use alternate paths rather than performing a new route discovery in the presence of route breaks. It further shows AODV-BR is not performing efficiently as the standard AODV with the increase of the traffic load. This is due to a higher number of packet collisions and contention due to the broadcast of data packets via alternate paths.

2.3.2 AOMDV: Ad hoc On-demand Multipath Distance Vector routing

AOMDV [26] [34] is an extension to the AODV protocol to provide multiple loop free link disjoint paths. The multiple paths are computed independently at each intermediate node by suppressing some RREQ copies and duplicating other RREQ copies. Each intermediate node keeps the maximum hop counts to the source and also the list of first path that it forwards. The protocol only allows accepting alternate reverse paths with lower hop counts compared to the maximum hop counts that it was set. In this manner,

the loop freedom is ensured. The destination replies to k copies which it receives from different neighbors. To ensure that paths in the route table are link disjoint, an intermediate node forwards the RREPs that do not have the same next hop and last hop as it is already in its route table. The RREP message carries the last hop detail which is the node immediately preceding the destination on that path. A RERR for a destination is generated when the last path to that destination fails like in AODV. Data is sent using one path at a time.

The results show that AOMDV offers reduction in end to end delay more than a factor of two. It provides 20% reduction in the routing overhead and the frequency of route discoveries.

2.3.3 AODVM: AODV Multipath routing

AODVM [35] is an extension to AODV for finding multiple node disjoint paths. All received RREQ are forwarded by intermediate nodes by keeping multiple route entries to the source in its RREQs table. The destination sends a RREP for all the received RREQ packets. An intermediate node forwards only a single RREP via the shortest reverse path to the source. When it forwards a RREP, it deletes all other entries in its RREQ table to make sure that it does not keep other routes to the source. Whenever an intermediate node overhears a RREP from a neighbor, it deletes that neighbor from its RREQs table. This makes sure that the node disjoint paths are created by keeping a single entry to the source and destination by an intermediate node. If an intermediate node cannot forward RREP further (i.e., no entry for the reverse path), it generates a RDER (Route Discovery Error) message and sends it back to the node from which it receives the RREP. The neighbor, upon receiving the RDER message, forwards the RREP to a different neighbor. Since intermediate nodes make decisions on where to forward the RREP messages, the destination is unaware about how many paths are created by the source. Therefore, a Route Confirmation Message (RCM) is sent by the source by piggybacking into the first data packet. Intermediate nodes are not allowed to send a route reply directly to the source.

The results are taken to analyze the discovery of number of node disjoint paths when varying the node density and the mobility in different topologies. There are few node disjoint paths are discovered for the networks with lower node densities. This paper further proposes a method to find reliable link disjoint paths since the probability of finding node disjoint paths is less in networks with lower node density. A method to find the locations to deploy reliable nodes in link disjoint paths is proposed based on the randomized min-cut algorithm.

2.3.4 MP-ODP: Multipath Routing for On-Demand Protocols

MP-ODP [36] proposes to discover alternate disjoint routes for the DSR protocol. Two methods are proposed. In the first, only the source gets multiple alternate routes. In the second, each intermediate node on the primary route gets an alternate route. In the first method, the destination replies to a selected set of RREQs. These routes should be link disjoint from the primary route. The primary route is the route taken by the first RREQ

reaching the destination. When the primary route breaks, the shortest remaining alternate route is used by the source. This process continues until all routes break and then a fresh route discovery is initiated. In the second method, the destination replies to each intermediate node in the primary route with an alternate disjoint route to the destination. Therefore, intermediate nodes make a decision about the alternate path to be used in case of link failure.

MP-ODP also provides an analytical modeling of the time interval between successive route discoveries for on-demand protocols based on a simple assumption on the lifetime of a single wireless link. This model shows that longer alternate paths are less advantageous, as they tend to break too early. Also, the performance advantage from using more than one or two alternate routes is usually minimal. It was shown in the simulation done for a network with 60 mobile nodes that MP-ODP has a better delivery rate, control overhead ratio, and error ratio, over DSR.

2.3.5 SMR: Split Multipath Routing

SMR [27] is an on-demand multipath routing protocol based on the DSR protocol. This is designed to utilize multipath simultaneously by splitting traffic onto two maximally disjoint routes. Two routes are maximally disjoint if they have a minimum number of common links. Unlike DSR, intermediate nodes do not reply to RREQs. This allows the destination to select maximally disjoint paths after analyzing all received RREQs. The multipath routes discovered with SMR do not guarantee that routes are node disjoint or link disjoint. The proposed route selection algorithm only selects two routes: the shortest delay route and the one that is maximally disjoint route. The destination sends RREP for the first RREQ it receives, which represents the shortest delay path. The destination then waits to receive more RREQs to select the next best path. If more than one maximally disjoint path exists, the path with the lowest hop count is selected.

The distribution of flows among SMR paths is done using a per-packet basis with round robin fashion. A new route discovery is started, after failing a single route or both routes. The simulation shows that SMR outperforms DSR in terms of delay and packet drops in an ad hoc network. SMR is more efficient when new route discovery is initiated only when both routes are broken, as it generates less control overhead.

2.3.6 MP-DSR: Multipath Dynamic Source Routing

MP-DSR [23] provides a multipath dynamic source routing protocol to improve QoS with respect to end-to-end reliability. The end-to-end reliability between the source and destination nodes is defined by evaluating the path reliabilities of all existing feasible paths. The path reliability is calculated based on the link availabilities of all the links along a path.

When an intermediate node receives the RREQ message, it checks whether it meets the predetermined path reliability requirement. If this RREQ message fails to meet such a

requirement, the intermediate node does not forward it further. When the destination receives the RREQ messages, it selectively chooses multiple disjoint paths from these messages, and sends RREP messages back to the source node via these selected paths. MP-DSR periodically checks the end-to-end reliability. The re-route discovery is initiated when either the reliability is no longer acceptable, or when all paths fail. The simulation results show that MP-DSR has better success delivery rate and less control overhead ratio than DSR.

Table 2-1 Review summary of existing multipath routing approaches

	AOMDV	AODV-BR	AODVM	MP-ODP	SMR	MP-DSR	OMR	DYMOM	ZDR
Utilization	APR	APR	APR	APR	SUM	APR/ SUM	SUM	APR	SUM
Basic Protocol	AODV	AODV	AODV	DSR	DSR	DSR	DSR	DYMO	LSR
Criteria for Route Discovery	link disjoint	partial disjoint	node disjoint/ link disjoint	link disjoint	maximally disjoint	node disjoint & path reliability	node disjoint, hcs, correlation factor	node disjoint	zone disjoint, hcs
Decision made at	I[1] & D	D[2] & I	I	D	D	D & I	S[3]	D	I
Check for Routing Loops	Forwarding few RREQs	By discarding duplicated RREQs	I nodes by overhearing RREPs	Source routing	Source Routing	Source Routing	Source Routing	path list	n/a[4]
Primary Path (P1)	SDP[5]	SDP	SDP	SDP	SDP	Highest reliable path	SDP	SDP	ZDR with lower hops
Flow Distribution	n/a	n/a	n/a	n/a	per packet splitting	per packet	n/m	n/a	n/m[6]
Re-route Discovery	when all fail	while using APR	n/m	when all fail	when both fail or first one fails	when all fail or path reliability is lower	when all fail	when all fails	when all fail
Route Maintenance[7]	Hello messages	by overhearing the data transmission on P1	n/m	in the route cache	with data transmission	in the route cache	with data transmission	Hello messages	with data transmission
Route Evaluation	no	no	no	no	no	yes	no	no	no
Implementation	NS-2 simulator	Glomosim simulator	NS-2 simulator	MaRS simulator	Glomosim simulator	Glomosim simulator	Glomosim simulator	NS-2 Simulator	QualNet simulator
Traffic generated	CBR	CBR	n/m	Exp. distribution	CBR	CBR	CBR	CBR	CBR
No: of Routes	3	Not limited	not limited	not limited	2	not limited	2-4	2	2
MAC Protocol	IEEE 802.11 with DCF	IEEE 802.11 with DCF	IEEE 802.11 with DCF	No MAC[8]	IEEE 802.11 with DCF	not mentioned	IEEE 802.11 with DCF	IEEE 802.11 with DCF	Modified MAC
Propagation model	n/m	Free space	n/m	n/m	Free space	n/m	Free space	Free space	n/m

[1] Intermediate node
[2] Destination node
[3] Source node
[4] n/a: not applicable
[5] SDP: Shortest Delay Path
[6] n/m: not mentioned
[7] When sending data over P1, approaches used to maintain the other path
[8] Error free wireless transmission without any multiple-access interference is used

2.3.7 OMR: On-Demand Multipath Routing for Mobile Ad Hoc Networks

OMR [37] introduces a multipath routing protocol to select paths based on three properties: (a) node-disjoint, (b) shorter paths (in terms of hop counts), and (c) small correlation factor between any two of the multiple paths. The correlation factor is defined as the total number of links that are within the communication range of each other. This is implemented by extending the DSR protocol. When the RREP message traverses from the destination to the source node, each intermediate node piggybacks the neighborhood information along the path. The source node calculates the path correlation factor using the neighborhood information piggybacked in the RREP message. The correlation factor is used only at the initial route discovery since the nodes are mobile and maintaining the correlation factor is costly in terms of control overhead.

2.3.8 DYMOM: DYMO Multipath Routing Protocol

The standard DYMO protocol [14] has been extended to keep multiple routes in [38]. DYMOM keeps only node disjoint routes. The destination keeps only two routes: they are the shortest path route and the route which has one more hop than the shortest path. Two paths are used only if they are node disjoint, by checking the node details that are carried in DYMO RREQ message. Data is initially sent via the shortest path only and the link breakages of the second path are detected with periodic Hello messages. In this way, the second path can be removed by generating RERR messages when it is no longer available due to node movements.

2.3.9 ZDR: Zone Disjoint Routes

In ZDR [39], a pair of paths are said to be zone disjoint if the data transmission over one path does not interfere with the other. Because omni directional antennas create unwanted interference in all directions, ZDR adopts directional antennas to reduce the overlapping area of the transmission range. The directional antenna used is called Electronically Steerable Passive Array Radiator (ESPAR). This is implemented based on a link state protocol by collecting information about the network topology. Each node periodically collects its directional neighborhood information through periodic beacons from each neighbor. A rotational sector based receiver oriented MAC protocol [40] is used to track the direction of its neighbors. Each intermediate node forwards the packets to a neighbor which has the lowest hop to the destination by making sure that it finds the shortest zone disjoint path. The ZDR is developed to discover SP or multiple paths with a maximum of 2 routes. If the hop counts are greater than 10, ZDR uses only the SP.

The average throughput is compared against a reactive AODV protocol with omni directional antennas. Though, the ZDR has higher overhead, results show that SP ZDR performs better than AODV even in mobile scenarios due to the use of directional

antennas. Furthermore, multipath ZDR performs better than single path ZDR providing more bandwidth.

2.3.10 Summary of Existing Multipath Routing Protocols

Table 2-1 compares the features of multipath routing used in different approaches that are discussed above. Apart from the multipath protocols detailed here, there are other approaches that can be used to create multipath such as TORA [41] and ROAM [42] protocols. Another routing concept called geographic routing which utilizes location information of each node when creating routes [43], is also a candidate protocol for multipath routing. Location information can be used easily to identify node disjoint routes though it carries more control overhead especially in mobile networks.

Furthermore, the above mentioned multipath routing research does not focus on utilizing multiple radios with different channel assignments. There are a few more research papers that discuss different channel assignments for different paths to avoid the interference between paths. They are mainly focused on modifying the hardware and software at the link layer [44] [45] [46] [47]. These proposals are not explained here since the investigations at the link layer to improve the performance of multipath routing is beyond the scope of this thesis.

2.4 Mutual Interference on Multipath Routing

A fundamental difference of wireless networks from wired networks is the mutual interference between links/paths located in proximity to each other. When multiple routing paths are interfering with each other, these paths cannot be operated simultaneously when using a shared medium such as IEEE 802.11 [48] technology. This phenomenon is known as route coupling and it restricts the possibility of the occurrence of simultaneous communications along the coupled routes. In general, routes that have nodes or links in common are considered highly coupled. However, route coupling may occur in wireless multi-hop networks, even if two routes have no common nodes or links.

Suppose two node disjoint paths as shown in Figure 2-3 are used to send data from S1 to D1 and S2 to D2 simultaneously. If the nodes in the S1 to D1 path and the S2 to D2 path are interfering with each other, these two paths cannot have simultaneous transmissions though they are physically separated. The data transmission delay of a path is not only dependent on the node characteristics of the nodes along the paths, but the interference from the neighboring nodes. The effect of route coupling is investigated in detail in the OPNET simulator as explained in the next section.

2.4.1 Evaluation of Interfering Routes

Figure 2-3 shows 3 independent node disjoint paths used for data communication from S1 to D1, S2 to D2 and S3 to D3. Each path consists of 5 ad hoc 802.11b nodes (with PHY mode set to 11 Mbps with RTS/CTS enabled) and routes are set manually to take the above paths. Left and right paths are not interfering while the middle path is inside

the interfering range of both left and right paths. As applications, a bidirectional video conferencing session at the rate of 840kbps and a single FTP download of a 10 MB file are used. The middle path experiences interference from the other two paths as shown in Figure 2-3.

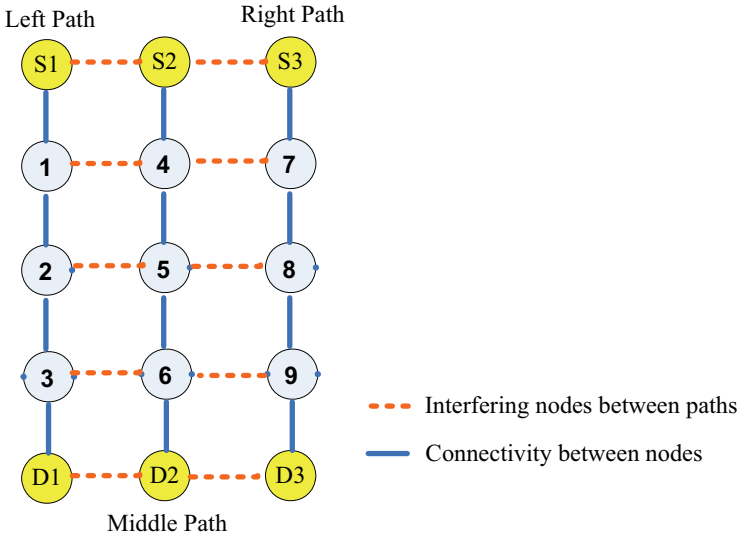


Figure 2-3 Simultaneous use of 3 independent node disjoint paths

The first test case, 3 individual video conferencing sessions are started at the same time over 3 paths. The next test case, 3 individual FTP downloads are started along the three paths at once. Table 2-2 shows the end to end delay and the FTP download response time, when using three paths simultaneously. This shows that the performance on the middle paths is worst, as it is interfering with other two paths simultaneously. The detailed analysis of results shows that the middle path hardly gets a chance to send data while data is transmitted on the other two paths. This phenomena is called the *flow in the middle problem* [16] in wireless multi-hop ad hoc networks.

Table 2-2 Simultaneous use of 3 node disjoint paths

	Left Path	Middle Path	Right Path
FTP: Download Response Time (sec)	1.016	5.244	1.065
Video Conferencing: average end to end delay (ms)	85.49	3120	85.11

The same test is repeated by avoiding the use of the middle path, while using only the left and the right path simultaneously. Here, the FTP download and the video conferencing session sent over middle path in the previous test now use the left path. Table 2-3 shows that the performance of applications that are sent over the middle path earlier have been significantly improved. Since the middle path does not carry any data, performance on the right path has also been improved for both applications. The performance on the left path has been degraded a little due to the increase of load by having two data streams. In summary, the overall performance of all the applications

have been improved significantly while avoiding the interference between simultaneously used multiple paths.

Table 2-3 Simultaneous use of 2 non-interfering node disjoint paths

	Left Path	Middle Path	Right Path
FTP: Download Response Time (sec)	1.860	x	1.005
	1.869		
Video Conferencing: average end to end delay (ms)	95.33	x	23.66
	94.92		

The above results show that the effect of route coupling becomes worst if the path is interfering more with the other paths. As a result, the nodes in the middle path are constantly contending to access the wireless medium and can end up performing worse than using one routing path at a time. This proves that node disjoint routes are not at all a sufficient criterion to improve the performance when using them simultaneously.

2.4.2 Overview to RDM Routes

As shown in section 2.4.1, the quality of transmissions may be degraded due to interference, even though the paths are physically separated (node disjoint). There are a few proposed metrics to measure the independence between the links of different paths. The correlation factor between two node disjoint paths is defined as the total number of links which are inside the transmission range of each other in [37]. The route coupling between two paths is calculated as the average number of nodes that are blocked from receiving data along one of the paths when a node in the other path is transmitting in [49]. The selection of routes having a lower correlation factor and route coupling provide better performance when using multiple routes simultaneously as proved by [37] & [49].

In summary, the selection of non-interfering routes is the main criterion to be addressed when using multiple routes simultaneously. This work introduces a new metric to select multiple routes by reducing the effect of interference between nodes as far as possible, which is termed Radio Disjoint Multipath (RDM). Probabilities of finding purely radio disjoint (no nodes in the interfering range of each other) paths are not always feasible in real wireless networks. However, keeping node disjoint paths with a minimum radio interference helps to avoid performance degradation (e.g. due to the flow in the middle problem in WLAN) up to some extent as shown in section 2.4.1. Furthermore, the path selection criteria have been extended to consider the existing Background Traffic Load (BTL) of a path together with the mutual interference between paths.

As mentioned before, though the paths are physically node disjoint or link disjoint, they may have mutual interference between links of different paths. Therefore, this work introduces 3 other types of multipath routing (Figure 2-4), assuming each node in the network uses the same wireless channel.

- Full Radio Disjoint Multiple Paths (FRDM): Mutual interference between all the intermediate nodes of each simultaneously active path is considered as zero. FRDM routes must be node disjoint.

- Partial Radio Multiple Disjoint Paths (PRDM): In this case, some of the intermediate nodes of each selected path are interfering while the rest are not. PRDM routes can be created either with link disjoint or node disjoint routes.
- Non Radio Disjoint Multiple Paths (NRDM): all the intermediate nodes of each selected path are interfering with each other. Even node disjoint routes can be NRDM depending on the topology of the nodes.

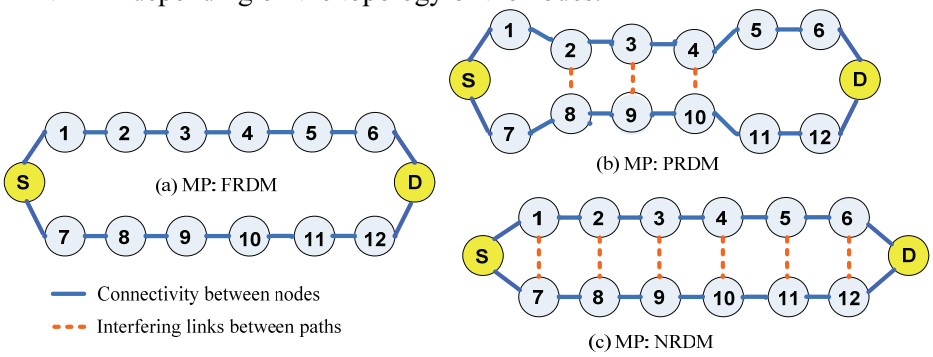


Figure 2-4 Types of RDM Routing: Full RDM, Partial RDM and Non RDM

The detailed description and the evaluation of RDM routes are explained in subsequent chapters. In summary, RDM routes introduce the following features compared to previous work.

- New path selection criteria for RDM routes instead of just discovering node disjoint or link disjoint routes
  - Consideration of mutual interference between paths
  - Consideration of background traffic load of each path
- Solution is based on the network layer: Algorithms proposed in this thesis to discover RDM paths, which can be implemented without any modifications to applications or link layer used (e.g. use of directional antennas, use of different channels). But, in the simulator, the computation of the number of packets at the MAC layer which measures the congestion and interference level of a node is done by modifying the link layer. In real environments, this can be done at the network layer (see section 3.4.1).
- Path evaluation: This introduces how to evaluate the paths due to the change of mutual interference with node mobility or due to the change of BTL of each path.
- Flow distribution based on SUM: This introduces 3 types of distribution methods: Multiple Flow (MF) distribution, Single Flow (SF) distribution and replicating packets of a single flow.
- Limit the SUM routes to 2: In order to find FRDM paths, paths should be node disjoint. In general it is difficult to discover a large number of disjoint paths in a network. Previous research has also proven that only 2 or 3 node disjoint paths provide better performance, when using them simultaneously [35] [36] and also the performance does not always improve by choosing the path with longer hop counts. The main objective of this work is to use RDM paths simultaneously.

Therefore this work limits the number of paths used to only 2 paths, which helps to avoid the flow in the middle problem and also to choose the best pair of paths with less interference.

3. Radio Disjoint Multipath Routing

This chapter explains the detailed operations of Radio Disjoint Multipath (RDM) routing. The first section details the RDM routing concepts, which can be applied to both reactive and proactive ad hoc protocols. The RDM routes are selected considering the mutual interference between paths together with the existing traffic load in a path. The second section details how these concepts can be applied for one of the reactive protocols, viz. the DYMO protocol. The third section gives an overview of the RDM routing implemented by extending the DYMO protocol. This implementation is done in the OPNET simulator. The fourth section discusses the feasibility of implementing the functions of the RDM routing in real multi-hop wireless ad hoc networks. The last section concludes this chapter, highlighting what features of RDM routing concepts are implemented in the OPNET simulator and looking at the possibility of implementing RDM concepts in a real implementation.

3.1 RDM Routing Concepts

The use of completely radio disjoint paths or paths with least interference is considered when using the RDM routing paths. Discovering the radio disjoint routing paths with no nodes in the interfering range of each other's path is not realistic. However, selecting node disjoint paths with minimum radio interference helps to avoid performance degradation (e.g. the *flow in the middle problem* in WLAN) to some extent as explained in section **2.4**. In addition to the interference between paths, the already existing traffic load in a path is also considered in the selection criteria. The discovered RDM paths are used simultaneously to distribute the traffic flows by utilizing the active routing paths simultaneously. The total number of paths that can be used to distribute the traffic flows is restricted to 2 since the use of more paths results in more mutual interference between paths.

The best pair of RDM paths must be selected by evaluating all the paths that are created during the route discovery process. The evaluation criteria are the mutual interference of a path and the Background Traffic Load (BTL) of a path. In order to measure the mutual interference, each pair of discovered paths has to be considered. This process requires an exhaustive comparison of each path against all the others that may require a high computational effort depending on the number of paths discovered. Equation 3-1 shows the total number of pairs of paths to be compared for n discovered paths. As an example, based on 3-1, for a discovery of 10 paths, a total of 45 pairs of paths have to be compared to compute the mutual interference of each pair.

$$\sum_{i=1}^{n-1}(n-i) \qquad n \geq 2 \tag{3-1}$$

Therefore, instead of utilizing the above mentioned exhaustive mechanism, a compromised mechanism is adopted. With this mechanism, the mutual interference of the paths is computed against a pre-selected reference path. The reference path, which is called the primary path, is selected based on 3 parameters: the interference of a path, the BTL of a path and the hop count of a path. The path with the least interference and the least congestion is selected as the primary path. Once the primary path is discovered, the next usable path, called the secondary RDM path is discovered by computing the total load of a path. The total load of a path consists of the mutual interference of a path computed w.r.t. the primary path and the existing BTL of a path.

The detailed description of how to find the pair of RDM paths (i.e. the primary and the secondary path) which have the least mutual interference and the least BTL is given in the following sub-sections. The RDM routing is developed by extending the standard messages of RREQ, RREP and RERR used in the wireless ad hoc protocols [12, 23, 49]. The following assumptions are made when developing the RDM routing for IEEE 802.11b based wireless multi-hop ad hoc networks. These assumptions are made by other wireless ad hoc protocols as well [1].

- Every node has a unique identifier. This can be the node's MAC address or IP address.
- All links are bidirectional. If a link exists from node i to node j, node j to node i transmission is also possible. If node k is interfering with node l, it is assumed that node l is also interfering with node k.
- The RDM paths are selected at the destination node by evaluating all RREQs received during a given time period. The RREQs received after this period are discarded assuming that the later RREQs are using more congested and interfering paths.

3.1.1 RDM Route Discovery Process

A network of a 5x5 grid topology is selected as shown in Figure 3-1 to explain the functions of the RDM routing in detail. In this topology, lateral neighbors are within their communication range and diagonal neighbors are not within each other's communication and the interference range. It is assumed that all the nodes are homogeneous w.r.t. their properties of transmission power, receiver sensitivity and other link layer parameters. The node S initiates the transmission of data to the node D.

The route discovery process of standard reactive protocols discards duplicate RREQ messages at intermediate nodes, as it is designed to keep only a single path [1]. Therefore, some of the possible paths to D might never be traced during the standard route discovery process. In order to find more paths, the RDM route discovery enables each intermediate node to forward all RREQs. In this process, the following criteria are enforced to have an implementation wise feasible and an efficient RDM route discovery process.

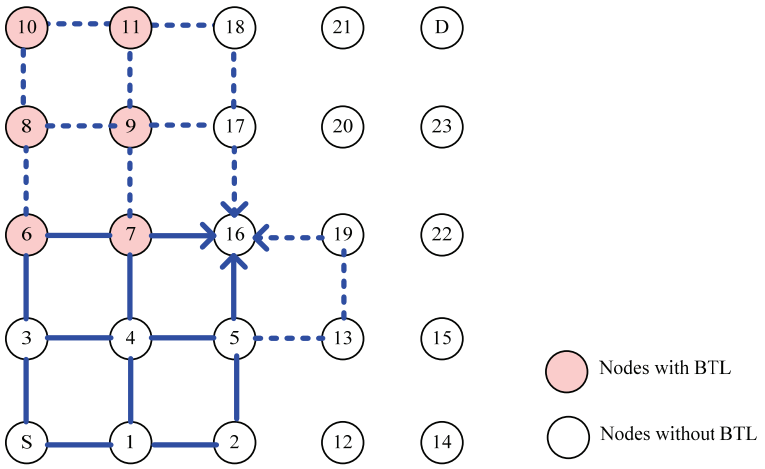


Figure 3-1 5x5 Grid topology: showing a propagation of RREQs via node 16

3.1.1.1 Detection of Routing Loops

Each intermediate node can avoid forwarding the RREQs that create routing loops by checking the node details of the created paths. Therefore, each RDM RREQ message should carry the path details as in the DYMO [24] and DSR [13] protocols. For example, node 16 in Figure 3-1 can hear a RREQ that was forwarded by itself previously, from node 17. Before processing this RREQ, node 16 should check whether its own identity is presented in the currently received RREQ message. In this way, node 16 can avoid the creation of routing loops.

3.1.1.2 Avoidance of Unnecessary Flooding of RREQs

Since the RDM route discovery allows each node to forward all RREQ messages, this could clog the network with more and more RREQ messages. When forwarding RREQs, each intermediate node creates the reverse routes towards S only for the selected RREQs. The RREQ is processed further if it satisfies 3-2.

$$HC_{current} \leq \min(HC_{existing}) + k \qquad 3\text{-}2$$

$HC_{current}$ refers to the hop count towards S over the currently received RREQ message. $HC_{existing}$ refers to the hop count towards S that has already being processed. The value of k can be changed for different network topologies and has to be configured to avoid unnecessary flooding of RREQs. Node 16 in Figure 3-1 should not process any RREQs that are forwarded by node 17 and node 19 if the value of k is set to 1. The assignment of lower values for k is based on the assumption that a path that can transmit the RREQs faster should be the least congested path and a longer path which has common

nodes with the already processed shorter paths does not support having more node disjoint paths.

3.1.2 RDM Path Selection Criteria

The RDM paths are selected by computing the interference and the BTL of each node in a given network topology. The following sub sections detail how the above criteria are measured during the RDM route discovery process.

In this thesis, a method is devised to measure the packets that traverse a node in terms of two parameters. The first measurement is to compute the number of packets transmitted by the node itself and received packets destined to the node itself. The second measurement is to compute all packets received from the others in the vicinity, but not destined to the node itself. The second measurement directly reflects the interference level of a node. These two measurements give a measure of the load and the interference level of a node, and are termed here as the Node Load (NL) and the Node Interference (NI) respectively. During the RDM route discovery process, the NL and the NI information can be propagated along the RREQ and the destination is able to use these values when selecting the RDM paths. The readings of these values should be computed for a number of samples taken for some time interval and taking the weighted average by assigning the higher weight to the latest readings.

In the absence of any existing traffic in a path, it is not possible to compute the NI at all as explained above. In order to measure the interference level of each node in the network, the NI has to be computed even in the absence of the BTL. The interference level of a node is required to find the primary path and then to compute the mutual interference of other paths. Therefore, two possible solutions are proposed to assess the interference of each node in the network.

- ***Solution 1 – Periodic Dissemination of Control Messages:*** Each node in the network has to periodically disseminate one hop broadcast messages even before the start of a route discovery process. From those messages, each node can compute the NI value easily. This method introduces more overhead to the network and also periodic dissemination of messages is not proposed for the reactive routing protocols. However, this solution is suitable and can easily be adapted to a proactive routing protocol [10] since each node in the proactive protocol disseminates control messages periodically even in the absence of data transmission. This solution requires measuring the NI by using the additional control messages when using reactive routing protocols.
- ***Solution 2 – Use of RREQs Messages:*** Since the RREQ messages are disseminated all over the network, these messages can be used to find the interfering neighbors of each node. For example, node 16 in Figure 3-1 can maintain its interfering neighbors upon receiving the RREQ message from the node 7, 5, 17 and 19. This list is called the Interfering Neighbor List (INL). Each node could attach its INL to a RREQ message in order to propagate all INLs to D. D uses all INLs to calculate the interference level of a node. Attaching the INL of each node increases the RREQ message. Another disadvantage of this

method is that a node is not able to complete its INL until it receives RREQs from its all neighbors. In order to avoid the above two problems, the completion of the INL for each node is proposed to be done at D as explained in section 3.1.2.2. This solution does not introduce additional overhead as in the previous solution as it uses existing RREQ messages to compute the interference of the network. However, the following two criteria have to be fulfilled to make sure that the computation of all the interfering neighbors uses this solution.

- Loss of a RREQ message may result in an incomplete INL. Theoretical and simulated computations of INL in two example networks are compared in section 3.4.2.
- All the interfering nodes of a particular node should physically be within a range that it could produce an interfering signal to the selected node. That means the receiving power at the selected node due to the other transmitting node should be greater than the receiver sensitivity of the selected node. All evaluated scenarios in this thesis use a PHY mode of 1 Mbps. Therefore, all the interfering nodes within the carrier sense range are computed in all discussed scenarios, since each node is configured with similar WLAN parameters such as transmitting power, receiver sensitivity, etc.

3.1.2.1 Computation of Background Traffic Load (BTL)

The computation of the BTL of an individual node is denoted as in 3-3. T_{ir} denotes the background traffic of the i^{th} node in the r^{th} path. It is considered as the sum of NL and NI of i^{th} node in the r^{th} path. The NL and NI are computed as explained in section 3.1.2.

$$T_{ir} = NL_{ir} + NI_{ir} \qquad 3\text{-}3$$

The computation of the BTL of a path can be denoted as in 3-4. T_r denotes the background traffic of the r^{th} path given as a percentage w.r.t the available node capacity. k_r represents the number of nodes in the r^{th} path. T_r is considered as the maximum load of a highly congested node in the r^{th} path.

$$T_r = \max(T_{1r}, T_{2r}, ...T_{ir}, ...T_{k_r}) \qquad 3\text{-}4$$

The use of maximum load of an individual node as the BTL of a path is justified with the simulation results by evaluating the throughput degradation in a SP when varying the BTL of different nodes (section 10.3). These results conclude that the degradation of throughput depends on the highly congested node in a path. If there are 2 paths having a highly congested node with the same amount of the BTL, then the path which has a higher accumulated BTL of all nodes give a lower throughput.

As mentioned in section 3.1.2, NL and NI are computed periodically by each node in the network. The RDM RREQ message carries two extra fields of 2 bytes each (see Figure 3-2) to carry the BTL information to D.

- $T_{\max}$ **:** keeps the maximum BTL of a path
- T_{acu} **:** keeps the accumulated BTL of a path

When S starts broadcasting a RREQ message, it attaches its own BTL (if it exists) to both fields of $T_{\max}$ and T_{acu}. Each intermediate node updates $T_{\max}$ carried by the RREQ message, only if it satisfies the condition shown in 3-5. T_{acu} is always accumulated by the BTL of the current node. In this manner, the maximum BTL of a node in a path is propagated along with the RREQ message in $T_{\max}$ and the accumulated BTL of all the nodes in a path in T_{acu}.

$$\begin{aligned} &if\ (T_{\max})_{RREQ} < (T)_{i\ node} \\ &\qquad (T_{\max})_{RREQ} = (T)_{i\ node} \\ &(T_{acu})_{RREQ} = (T_{acu})_{RREQ} + (T)_{i\ node} \end{aligned} \qquad 3\text{-}5$$

Upon receiving the RREQ message with both $T_{\max}$ and T_{acu} values, D computes the BTL of the r^{th} path, T_r which is considered as the $T_{\max}$ value in the RREQ received over the r^{th} path. If D finds more than one RREQ that gives the same BTL (i.e. same values of $T_{\max}$), D should further process these RREQs to find the least congested path based on T_{acu} according to 3-6. Here T_r is normalized w.r.t. total values of T_{acu} in both paths.

$$\begin{aligned} &if\ (T_i = T_j)\ \&\ (T_{acu})_i \le (T_{acu})_j \quad where\ i \neq j \\ &\qquad set\ T_i = (T_{acu})_i / \{(T_{acu})_i + (T_{acu})_j\} \\ &\qquad\quad T_j = (T_{acu})_j / \{(T_{acu})_i + (T_{acu})_j\} \\ &if\ (T_i \le T_j) \\ &\qquad set\ T_i\ as\ "least\ congested"\ path \\ &else \\ &\qquad set\ T_j\ as\ "least\ congested"\ path \end{aligned} \qquad 3\text{-}6$$

3.1.2.2 Computation of Interference

The measurement of the interference level of each node in a network is done either using the dissemination of additional control messages or using the existing RREQ messages as explained in section 3.1.2. This section details how to assess the interference level of each node using the existing RREQ messages. This requires D to complete the INL of each intermediate node as explained above. The INL is used to

select the primary path, P_1 and then to compute the mutual interference of other node disjoint paths w.r.t. P_1.

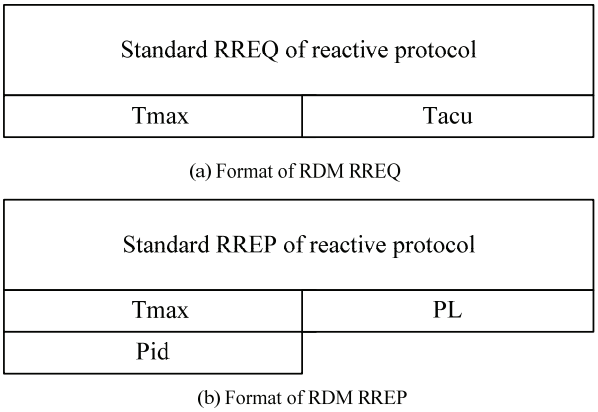


Figure 3-2 Format of RDM RREQ and RREP messages

Assuming that D receives two RREQs that include node 16 as an intermediate node:

- via nodes 3, 6, 7, **16**, 17, 18, 21
- via nodes 1, 2, 5, **16**, 19, 20, 21

From the above two RREQs, D completes all interfering nodes of node 16 assuming all links are bidirectional and no RREQs are lost during the propagation. Therefore, the INL of node 16 consists of {node 7, node 5, node 17 and node 19}. In this manner, D can complete the computation of the INL for each node which is included in all the paths that are received by D.

3.1.2.2.1 Selection of the Primary RDM Path

The primary path P_1 is selected avoiding paths which have a higher number of interfering nodes, a higher BTL and a higher number of hop counts. The number of interfering nodes is computed by accumulating the total number of nodes that are interfering in a path as shown in 3-7.

$$I_r = \sum_{i=1}^{k_r} sizeof(INL_i) \qquad 3\text{-}7$$

I_r refers to the total number of interfering nodes in the r^{th} path. This is computed by accumulating the total number of nodes in all INLs of the r^{th} path. The details of 4 selected paths through which RREQs reached D for the topology in Figure 3-1 is shown as follows.

- P_1 = via node 1, node 2, node 12, node 14, node 15, node 22 and node 23

- P_2 = via node 3, node 4, node 5, node 13, node 19, node 20 and node 21
- P_3 = via node 3, node 6, node 7, node 16, node 17, node 18 and node 21
- P_4 = via node 3, node 6, node 8, node 10, node 11, node 18 and node 21

Table 3-1 shows that the complete INLs of both P_1 & P_2 together with total number of interfering nodes in each path. Similarly, the total interfering nodes of P_3 and P_4 can also be computed as $I_3 = 24$ and $I_4 = 20$ respectively. D can select either P_1 or P_4 as the path which has the least interfering nodes. Since both paths have same hop counts and P_4 is carrying the BTL as shown in Figure 3-1 (node 6, 8, 10 & 11 are being used for the BTL), the D must select P_1 as its primary path in this situation.

Table 3-1 Details of all INLs in P_1 and P_2

	node i	(INL)i	Size of (INL)i
P_1	node 1	{S,4,2}	3
	node 2	{1,5,12}	3
	node 12	{2,13,14}	3
	node 14	{12,15}	2
	node 15	{14,13,22}	3
	node 22	{15,19,23}	3
	node 23	{22,20,D}	3
I_1			***20***
P_2	node 3	{S, 6, 4}	3
	node 4	{3,1,7,5}	4
	node 5	{4,2,16,13}	4
	node 13	{5,12,19,15}	4
	node 19	{13,16,22,20}	4
	node 20	{17,19,23,21}	4
	node 21	{18,20,D}	3
I_2			***26***

3.1.2.2.2 Selection of the Secondary RDM Path

Since RDM paths have to be node disjoint, D should not further process the RREQs that create non disjoint routes w.r.t. P_1. Then D should compute the mutual interference only for the selected node disjoint paths w.r.t. P_1. The mutual interference index of the r^{th} path w.r.t. P_1 is denoted as I_{1r} and can be computed as given in 3-8.

$$I_{1r} = \sum_{j=1}^{k_1} \sum_{i=1}^{k_r} sizeof\,(A \cap B) \quad where\ A = \{n_j\}\ \&\ B = \{INL_i\} \qquad 3\text{-}8$$

k_1 refers to the total number of nodes in P_1. n_j denotes the j^{th} node of the primary path of P_1. $(INL)_i$ refers to the set of all interfering nodes in the i^{th} node of the r^{th} path. The

set A gives all the nodes in P_1 and the set B shows all the nodes in the i^{th} INL of the r^{th} path. In other terms, mutual interference w.r.t. P_1 is computed by counting the total number of nodes in P_1 which are interfering with the nodes in the r^{th} path. For example the mutual interference between P_1 & P_2 can also be shown as in Table 3-2. Similarly, mutual interference of P_3 & P_4 w.r.t. P_1 can be computed as $I_{13} = 0$ and $I_{14} = 0$ respectively. Both paths P_3 and P_4 do not have any interfering nodes from the primary path.

Table 3-2 Computation of mutual interference of P_2 w.r.t. P_1 (I_{12})

Nodes in P_1 *(A)*	(INL)i of P_2 *(B)*	Interfering nodes of P_1
node 1	node 3: {S, 6, 4}	-
node 2	node 4:{3,1,7,5}	node 1
node 12	node 5:{4,2,16,13}	node 2
node 14	node 13:{5,12,19,15}	node 12 & node 15
node 15	node 19:{13,16,22,20}	node 22
node 22	node 20:{17,19,23,21}	node 23
node 23	node 21:{18,20,D}	-
I_{12}		***6***

By knowing the mutual interference and the BTL of each selected node disjoint path, The total load of a path, called as (*PathLoad, PL*) can be computed as follows. *PL* is computed by combining I_{1r} and T_r as shown in 3-9. The weight factor α can have values between 0 and 1. The path with the least *PL* is chosen as the secondary RDM routing path. When several paths have equal *PL*, the path with the lower number of hops is chosen. m is the total number of node disjoint paths that are discovered w.r.t. the primary path.

$$PL_r = \alpha I_{1r} + (1-\alpha) T_r \qquad 1 < r \le m \qquad \text{3-9}$$

Assuming that the computation of *PL* for the above selected node disjoint path combinations in Figure 3-1, Table 3-3 shows the computed PL_2, PL_3 and PL_4 for two different values of α. In this computation, the interference index has been normalized w.r.t. the maximum mutual interference of a path, i.e. $I_{12} = 6$. Assuming that nodes 6, 8 and 10 are transmitting 100 bytes/sec to nodes 7, 9 and 10 respectively, the BTL of P_3 and P_4 can be computed according to 3-4. T_2 is equal to the BTL of node 3 which consists of only NI of 100 bytes/sec that is heard from node 6. T_3 is equal to the total BTL of node 6 which consists of NL of 100 bytes/sec and NI of 100 bytes/sec. The NI at node 6 is equal to the total of all packets that are heard from node 8. Similarly, T_4 is

equal to the total BTL of node 8 consisting of 100 bytes/sec of its own NL and the NI of 200 bytes/sec that are heard from both node 6 and node 10. The normalized values of the BTL of P_2 , P_3 and P_4 are computed as $T_2 = 0.33$, $T_3 = 0.66$ and $T_4 = 1$ respectively.

Table 3-3 Computation of PL_2 , PL_3 and PL_4

α	PL_2 $I_{12} = 1 \& \ T_2 = 0.33$	PL_3 $I_{13} = 0 \& \ T_3 = 0.66$	PL_4 $I_{14} = 0 \& \ T_4 = 1$
0.7	0.799	0.198	0.300
0.2	0.464	0.528	0.800

Table 3-3 shows that the value of α has to be weighted more to represent the behavior of mutual interference. When α is set to 0.7, P_3 gives the lowest PL and can be chosen as the other RDM path to be used with the primary path P_1. The pair of P_1 and P_3 is the least interfering and least congested route in this topology. P_2 can be selected as the other path if using the lower value for the weight factor α. In this case, the congested paths can be avoided rather than considering the mutual interference between the paths. Chapter 5 details how the performance of applications can be affected, when choosing different types of RDM paths.

Further, if the secondary path is more congested (having higher PL irrespective of the value of α), the RDM protocol should not select the secondary path. In this situation, using the primary path alone may enhance the performance of applications compared to the use of 2 interfering routes simultaneously.

3.1.3 Sending of RDM RREP

Once D decides which pair of paths to be used, D sends RREP messages for both selected paths as in the standard reactive protocol. As shown in Figure 3-2, it carries additional 3 fields.

- ***Tmax:*** keeps a maximum BTL in the selected path.
- ***PL:*** keeps a computed PL of the selected path as in 3.1.2.2.2
- ***Pid:*** keeps an identification number given to a path. Pid is kept in the routing table to identify a path uniquely. Though D sends the RREP via P_1 first, there is no guarantee that S also receives the RREP via P_1 first. Therefore, pid is introduced to identify the path uniquely at S and D. When distributing traffic flows, if S decides to send some flows to the primary path, D should also use the same path to redirect the packets of the same flows, when using bidirectional flows. Otherwise, the performance of applications can degrade due to the use of non identical paths to distribute packets of the same flow. For example, if TCP acknowledgements are sent via a path with a lower delay while TCP data is sent via a path with a higher delay, the TCP sender could compute a lower value for the average RTT than the actual time taken to send TCP data via the path with

the higher delay. This causes premature TCP timeouts and incorrect increase of congestion window size based on wrong estimates. This results in overloading the path with the higher delay. In case of UDP traffic, propagation delay between two paths can create higher variations in jitter.

Table 3-4 Details of RDM RREP message

	NLmax	PL	Pid
RREP sent via P_1	0	0	1
RREP sent via P_3	0.6	0.15	2

3.1.4 RDM Flow Distribution Criteria

The motivation of discovering the RDM paths is to use both RDM paths simultaneously to distribute the traffic load. The simultaneous use of RDM paths by traffic flows can be done in the following manner as explained in Chapter 1.

- **Multiple Flow (MF) distribution:** Distribution of independent flows among RDM paths.
- **Single Flow (SF) distribution:** Distribution of packets of a flow among RDM paths. This can be used in two ways. Firstly, by splitting/distributing packets of a flow among multiple routing paths to balance the load in the network. Secondly, replicating packets of a flow to all active paths.

In both distribution methods, each individual flow has to be identified uniquely in order to identify a packet of a particular flow to be distributed. As explained in section 2.2.2.1, there are different distribution algorithms to distribute flows based on different criteria. These criteria in general could be the mapping of a flow to a given path based on properties such as bandwidth, delay, QoS, etc. [27, 29]. The distribution criteria to be used with the RDM paths do not depend on the RDM protocol. Therefore, any distribution algorithms and criteria can be implemented when using the RDM paths simultaneously. The distribution algorithms and criteria used in this work are explained in detail in section 5.2.

The implementation of the above two distribution methods requires addressing of the following issues:

- When distributing packets, the packets that belong to the same flow have to be identified uniquely at the IP layer. There are different ways of identifying flows by using information in the IP header such as protocol, port numbers, DSCP (Differentiated Services Code Point), IP addresses and so on.
- When splitting packets, there is a possibility that packets are received out of order. This causes TCP to react negatively and higher jitter is experienced in audio/video applications. Therefore, distribution criteria must be designed to avoid out of order packet delivery as much as possible.

- When replicating packets, there is a possibility that the same copy of the packet is received more than once. When TCP sends TCP-ACK for all redundant copies, the TCP sender enables fast retransmission for the receipt of replicated TCP-ACK. This reduces the TCP throughput unnecessarily. Therefore, the destination should discard the redundant copies when replicating packets at the sender [32].

3.1.5 RDM Path Maintenance

In the RDM routing, D and S nodes attempt to keep multiple routes while the intermediate nodes maintain only a single path. When utilizing multiple routing paths simultaneously, path lifetimes can be updated as in the standard reactive routing protocol. This is done by extending a route lifetime upon receiving any control packet, data packet or link layer acknowledgments. If an intermediate node detects that there is no data transmission during a predefined interval, it determines that a neighboring node is not reachable. Then, it should broadcast a RERR message. This is done only, if it has an active route via the unreachable neighbor. Upon the receipt of a RERR message by S or D, it first deletes the path of concern (i.e. path indicated by the RERR). A re-discovery of routes is initiated after reevaluating the network as in section 3.1.6.

S and D should extend the path lifetime even when sending a packet in addition to the arrival of data packet in the standard way. The update of lifetime when sending a packet has to be done selectively by identifying the path in which the data packet has been forwarded. For example, if S distributes traffic among P_1 and P_3 in Figure 3-1, S should update the next hop of node 1 if it forwards data via P_1 and the next hop of node 3 is updated when forwarding data via P_3.

3.1.6 RDM Dynamic Path Evaluation

Once S and D decide the RDM routes to be used as explained in section 3.1.2, these routes are utilized to transmit data simultaneously until there is no route failure or no necessity to change the chosen routes. In case of a route failure, S or D should be informed with a RERR message (see section 3.1.5). On the other hand, the RDM protocol should evaluate the discovered routes dynamically if there is a change in the BTL or the interference in the network.

The RDM dynamic path evaluation process requires the use of new control messages, but without introducing more overhead in the network as follows. The path evaluation process should not be initiated if the existing active route performs satisfactorily and there are not many flows to be distributed.

The following messages are introduced to evaluate the RDM paths dynamically. These messages are designed similar to standard on-demand control messages, but with an additional flag to distinguish them from the standard messages.

- ***Path Evaluation RERR (PE-RERR):*** contents in this message is similar to the standard RERR message, but with an additional flag set to identify PE-RERR uniquely.
- ***Path Evaluation RREQ (PE-RREQ):*** contents in this message is similar to the RDM RREQ message, but with an additional flag set to identify PE-RREQ uniquely.
- ***Path Evaluation RREP (RE-RREP):*** contents in this message is similar to RDM RREP, but with an additional flag set to identify PE-RREP uniquely.
- ***Path Evaluation Confirmation (PE-C):*** this is a hop by hop forwarding message to confirm the use of newly discovered routes.

In general, each intermediate node of the discovered path should notify S or D if there is a change in the BTL or the break of routes due to node mobility or a node failure. If S does not get any notifications from the intermediate nodes, S should initiate the path evaluation process after some period. The detailed evaluation process is explained using Figure 3-3, assuming that the already discovered RDM paths for the topology in Figure 3-1 are P_1 & P_3 as explained in section 3.1.2.

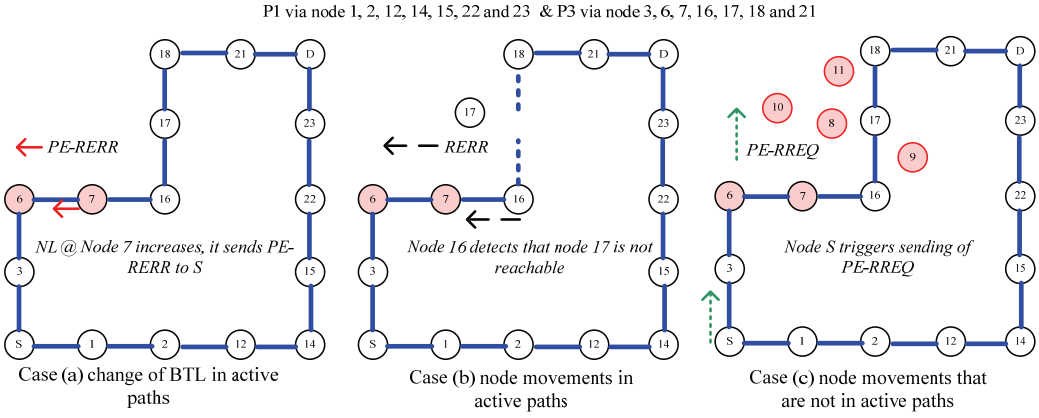


Figure 3-3 Change of the BTL and the network topology while distributing traffic among the RDM routes of P1 and P3. The shaded nodes are configured with the BTL

The RDM dynamic path evaluation should be initiated due to one or more of the following changes in the network.

- There is a change in the BTL of the nodes in the already discovered active paths. This is similar to the case (a) of Figure 3-3 where the BTL of node 7 increases to a level above the previously measured BTL of P_3. In this case, node 7 should send a PE-RERR message to S to inform that its existing BTL has been changed. It places its own IP address to the "*unreachable destination*" field of the PE-RERR message. The BTL is computed by using the NL and the NI as explained in section 3.1.2, with considering only the packets that are generated due to the

BTL. An intermediate node processes the PE-RERR as a standard RERR, but without deleting the routes for the "*unreachable destination"* of the PE-RERR message. The unreachable destination in the PE-RERR message carries the IP address of a node which has a higher BTL than earlier. Upon receiving the PE-RERR, S or D should initiate the path evaluation process.

- There is a topology change as in case (b) of Figure 3-3, where the intermediate nodes of active paths are moving (i.e. node 17). In this case, node 16 should send a RERR message after detecting that the link with node 17 is broken. Upon receiving a standard RERR message, S should redirect the data packets on the broken paths to the other active path until S evaluates the network to find better RDM paths.
- There is a topology change as in case (c) of Figure 3-3, where the intermediate nodes that are not part of the active paths are moving. In this case, there is no possibility that S or D is informed as in earlier cases about the change of network conditions. Therefore, S should always initiate the path evaluation after some period in order to reevaluate the paths.

Upon receiving a PE-RERR or a RERR message, S should send a PE-RREQ to evaluate the network conditions again. If D finds a better pair of RDM paths during the PE-RREQ propagation, D sends newly discovered path details by unicasting a PE-RREP. The processing of the PE-RREQ and the PE-RREP messages are similar to the processing of the RDM RREP and RREQ messages except updating the routing table. Since the evaluation process should not modify the existing active path details, nodes should keep the routing information in a separate routing table called a Path Evaluation Routing Table (PE-RT) during the processing of PE-RREQ and PE-RREP messages. Once S decides to use the evaluated path, S should send a confirmation message to set the new routes in the PE-RT. This is done using a PE-C message, which is forwarded hop by hop by identifying the next hop from the PE-RT. All nodes set new routes to both reverse and forward paths during the propagation of the PE-C message. This message carries the addresses of both S and D together with their own sequence numbers to identify routes uniquely for a given pair of S and D.

The newly discovered paths have to be established first and the previously used RDM paths have to be deleted after making the new routes. The deletion of paths can easily be done by not distributing flows to the bad routes. Without any data transmission, the path lifetime is expired and those routing paths are removed permanently.

Furthermore, the path evaluation process must ensure that there should not be unnecessary transmissions of PE-RERR and PE-RREQ messages in the network. Therefore, the following methods are proposed to make the route evaluation process more efficient.

- An intermediate node should send a RE-RERR message only after the current BTL level exceeds a threshold of the earlier computed BTL of a path.
- S should not initiate the path evaluation process if the existing active route is satisfactory and there are not many flows to be distributed.

- D should not set up new RDM paths if the BTL of the existing active route is comparable with the PL of newly discovered routing paths in terms of mutual interference and the BTL.

3.2 RDM Routing based on DYMO

The functions of the RDM protocol are realized in a simulated environment by extending the DYMO protocol. This section gives an overview of the DYMO protocol.

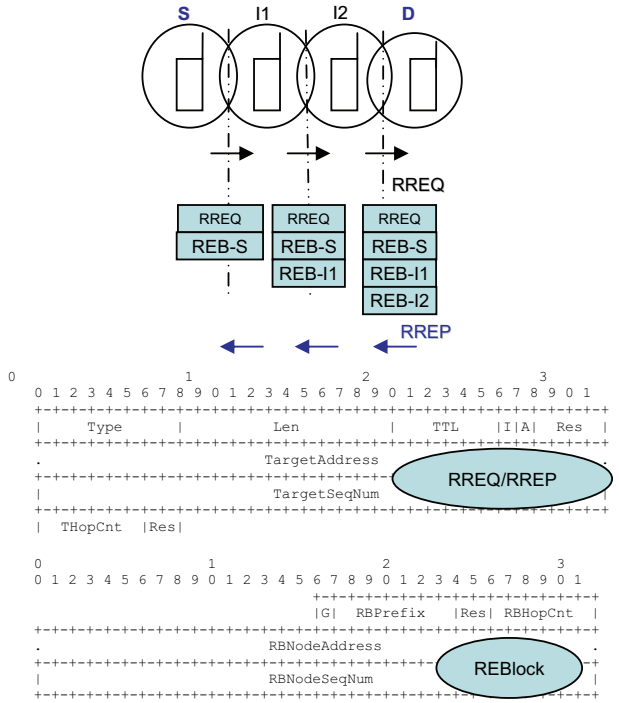


Figure 3-4 Format of DYMO RE message consists of RREQ/RREP & REB

Currently, the IETF MANET working group [9] focuses on standardizing a reactive MANET protocol, which has both good properties of the AODV [12] and the DSR [13] experimental protocols. The DYMO protocol [24] is the candidate solution that is being discussed at the IETF. The mechanism of DYMO's route discovery is basically the same as its counterpart in AODV, such as flooding RREQ from S and unicasting RREP by D. But in comparison to AODV, the RREQ and RREP messages in DYMO have different message formats, which are named Route Element (RE), which consists of RREQ/RREP and Route Element Blocks (REB) details, as shown in Figure 3-4. The major difference between AODV and DYMO lies in how the Intermediate (I) nodes process the REBlock. While propagating REs among the intermediate nodes (i.e. I1 & I2), each intermediate node sets up a reverse path to node S upon the reception of the

first copy of the RE message. This is the same for both AODV and DYMO. However, in DYMO, there is an option for each intermediate node to append its detailed route information to the RREQ by adding its own REBlock to the end of the RREQ. Therefore, the RE that reaches D contains the route information of each node along a given path. This feature of DYMO is called **path accumulation** and it alleviates time delays introduced when route discoveries originate from I1 or I2 since they all have routes to each other in the active path [50].

3.2.1 RDM aware DYMO Routing

The RDM protocol can easily be implemented using the standard DYMO protocol as DYMO RE message carries each intermediate node details in its REBlock. The existing DYMO RREQ is extended to carry $T_{\max}$ and T_{acu} of a path as explained in section 3.1.2.1.

Since the RDM routes have to keep multiple next hops for the same destination, the routing table of standard DYMO has been extended as shown in Figure 3-5. Multiple next hops are kept in a list called *"NextHopList"*, where each entry has the hop counts, computed PL, $NL_{\max}$ together with an id given to a selected path. The *OrginalBW*, *RemainingBW,* the *FIDList* and the *SentCount* are used when distributing traffic based on the different criteria (see section 5.2.1).

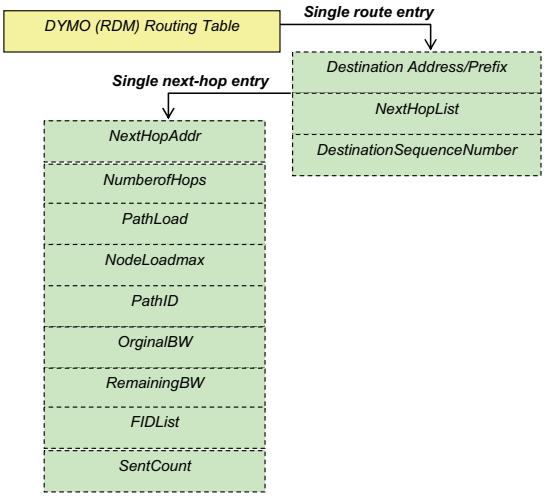


Figure 3-5 Extended DYMO routing table for RDM

The D and the S nodes attempt to keep multiple routes while the intermediate nodes maintain only a single path. When utilizing multiple RDM paths, path lifetimes are updated based on the DYMO protocol (i.e., based on signaling packets and data packets that traverse a path).

3.3 Implementation of RDM Routing in the OPNET Simulator

This section gives an overview on how the RDM aware DYMO protocol has been implemented in the OPNET simulator [51].

OPNET is a discrete event simulator used in simulating networking environments. It is one of the widely used simulators by the research community. It contains a large number of models that can simulate protocols from higher levels to lower levels of the protocols stack. OPNET at a very high level of abstraction consist of 2 parts: Simulator Core and Protocol Implementations.

The core of the simulator provides the base functionality like any simulator such as event generation. This part of the simulator is closed for any extensions by the users. The protocol implementation part of the simulator, where the actual functionality related to implemented protocols resides, is open to the users of OPNET. This means that any user can modify the behavior, extend or even implement completely new protocols in OPNET.

OPNET organizes its environment into a number of components in a hierarchical manner. These components are Scenarios, Node Models and Process Models.

- The scenarios consist of networking elements that are brought together to simulate the behavior of a network or part of a network. A scenario consists of a number of nodes. A node is the representation of a network element with all the networking functionality required for a given scenario. An example of a node is a mobile node capable of performing wireless communications.
- The node and the functionality are referred to as the Node Model.
- A Process Model implements the actual functionality of the different components that make up a node. This functionality is represented in terms of state transitions. An example of a Process Model is the implementation of the TCP protocol which is a transport layer protocol in TCP/IP suite.

In the framework of this thesis, the RDM aware DYMO protocol has been implemented in the OPNET simulator by modifying 3 major layers/processes, as shown in Figure 3-6 [52].

- The first change relates to extending the current DYMO implementation (based on the IETF DYMO draft 02). These modifications are done in the child process of the **manet_rte_mgr** process model that handles all DYMO operations which is called the **dymo_rte** process model.
- The second set of changes is done at the IP layer to implement all distribution algorithms. These changes are done in the **ip_rte_central_cpu** process model and the **ip_cmn_rte_table** external file. The communication between **dymo_rte** and **ip_cmn_rte_table** external file is done by introducing a global structure

*(***ip_cmn_rte_table** obtains the list of routing paths from the **dymo_rte** process model that must be used for distributing packets).

- The third set of changes is done to get the NL and the NI readings in **wlan_mac** process model. This implementation requires inter process model instance communication to pass information and this is handled using the functions of the Internal Model Access (IMA) package in OPNET, i.e., the communication *between* **dymo_rte** *and* **wlan_mac** process models (**dymo_rte** process model obtains the NL of the wireless network interface computed periodically from the **wlan_mac** process model).

3.3.1 DYMO Process Model

Processing of the DYMO RE messages are separated based on the user defined attribute of enabling/disabling the RDM paths, in the DYMO process model. To handle multipath routing (i.e. RDM aware DYMO), a new set of parameters is added to the standard **dymo_rte** process model:

- Enable RDM: Enables/disables the RDM feature in the DYMO. If disabled, standard (i.e. single path) DYMO is used.
- Distribution method: Identifies the way in which the RDM paths are used. The possible values are “MF”, “SF-Splitting” and “SF-Replicating”. MF distributes individual flows to the selected path. SF-Splitting splits the packet flow into the selected paths while SF-Replicating makes multiple copies of a single packet and then distributes it to the selected paths.
- NL and NI Computation Parameters: The “Retrieval Period” specifies the duration in seconds after which the values that are used to compute the NL and the NI are read by DYMO from the WLAN layer. The “Num Readings” specifies the number of consecutive NL and NI values to be held to compute a weighted average of these readings.
- RREQs Waiting Time: This defines the time period that D should wait to evaluate all the received RREQ messages.

The RDM aware DYMO process model is extended with additional functions of:

- ***RDM Route Discovery:*** Upon receiving the DYMO RE, each intermediate node attaches new REBlock details for each valid RE to be processed. Validity of the RE is checked based on the standard DYMO logic by comparing sequence numbers without considering the hop counts. The RE message is re-broadcasted only if it satisfies 3-2. Before re-broadcasting the RREQ message, each intermediate node computes its current BTL as explained below and updates the Tmax and Tacu fields of the RREQ. After the expiry time as set in “RREQs Waiting Time” attribute, D selects the primary path and the secondary path as explained in section 3.1.2. If there is no secondary path, which is node disjoint with the primary path, D sends the RREP via only the primary path. D sends the RREP with computed details of PL, Tmax and Pid.
- ***Check for Disjoint Routes:*** The basic idea of this functionality is to determine paths which do not have any common nodes with the primary path. If it contains at least one common hop, that path is considered as a non disjoint path and disregarded. To perform this task, the search code that is implemented consists

of 2 nested loops that iterate through the node details of the REBlocks of the primary path and the path to be checked.

- ***Maintenance of multiple routes:*** D and S nodes attempt to keep multiple routes while the intermediate nodes maintain only a single path. When utilizing multiple routing paths, a path lifetime is updated based on the DYMO protocol (i.e., based on signaling packets and data packets that traverse a path). Upon the receipt of a RERR message by the S or D, it first deletes the path in concern (i.e. path indicated by the RERR). A re-discovery of routes is not initiated if there is at least one active path in the NexthopList.

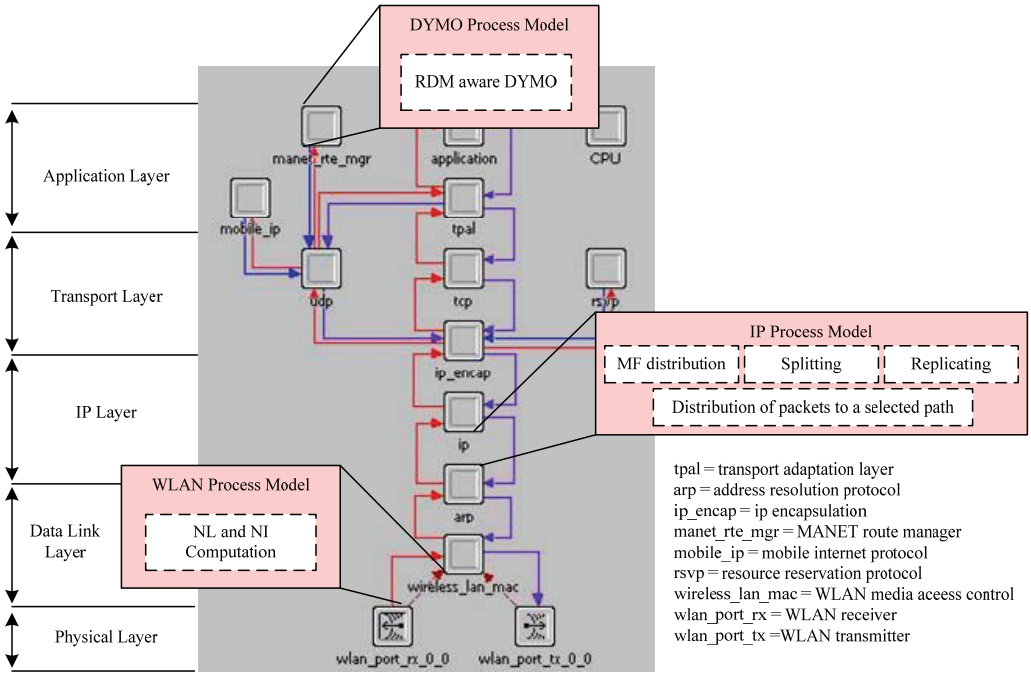


Figure 3-6 Locations of extended functionality (RDM aware DYMO, NL and NI computation, distribution methods) in the node model of the OPNET Simulator: manet_rte_mgr is capable of accessing the transport layer (to send UDP based control messages) and the IP layer (to setup routes)

3.3.2 Retrieval of NL and NI the WLAN MAC layer

The **wlan_mac** process model is changed to assist the computation of the NL and the NI for each node in the network. The NL is computed by capturing packet sizes of packets that traverse through the MAC layer of a node. It holds 3 state variables that keep track of the following types of packets (i.e. processed by the MAC layer).

- X = Packets destined to the node
- Y = Packets originated by the node
- Z = Packets destined to other nodes

These variables are held in byte units though they are processed in the **wlan_mac** process model as bits. The readings are accumulated every time a packet is picked up belonging to any one of the above categories. The **wlan_prepare_frame_to_send** function contains the hook to obtain sizes of the outgoing packets while the **wlan_physical_layer_data_arrival** function contains hooks to get the sizes of the incoming packets that are destined to the node itself and to the other nodes as well. The incoming packets are considered only if the receiving power is greater or equal to the receiver sensitivity at the receiving node. These 3 state variables are read by the **dymo_rte** process model periodically and initialized after every read.

t=2 NL2	t=3 NL3	t=4 NL4	t=5 NL5

@ t = 5 sec, Weighted NL = (0.4 *NL5) + (0.3 *NL4) +(0.2 *NL3) +(0.1 *NL2)

Figure 3-7 Computation of NL based on weighted average

In the **dymo_rte** process model, both X and Y variables are added to compute the NL while the Z variable is directly considered as the NI. As explained in section 3.1.2.1, both NL and NI are read periodically and the weighted average is computed. As shown in Figure 3-7, the weighted average of NL is computed when setting the "Retrieval Period" to 1 sec and "Num Readings" to 4. A periodic retrieval of these values is kept in a temporarily array, which is used to compute the weighted average in a way where a higher weight is assigned to the latest readings of the NL and the NI. The RREQ message carries the computed weighted average as explained in section 3.1.2.1.

3.3.3 Distribution of Flows at the IP Layer

S and D distribute traffic flows based on 3 mechanisms, i.e., distributing independent flows, splitting packets of a single flow, replicating packets to each path. The changes to the IP layer functionality consist of the following.

3.3.3.1 Multiple Flow Distribution

MF distribution is implemented by identifying a flow based on ToS (Type of Service) value used in the OPNET applications. The ToS is similar to a DSCP value in IPv4 packets. Uniquely identified flows are assigned to different paths based on the criteria explained in section 5.2.1.

3.3.3.2 Splitting of IP packets

Splitting of packets (i.e. distribution) is done in **ip_cmn_rte_table** external file. The new code uses the round robin mechanism to distribute packets to the usable next hop of the respective route. The algorithms used to decide how many packets are sent to a selected path at once are discussed in section 5.2.2.

3.3.3.3 Replication of IP packets

Replicating each packet at the originator is implemented together with discarding of redundant packets at D. The actions of replication and discarding multiple copies are done in the **ip_rte_central_cpu** process model.

The function **ip_rte_central_cpu_packet_arrival,** handles the packets that come to the IP layer from different sources. These are:

- Packets arriving from the lower layers
- Packets arriving from the upper layers
- Packets that are internally communicated (within IP layer)

Replication is done for packets that arrive from the upper layers, while discarding replicated packets is done for packets that come from the lower layers. Discarding of packets is done based on the *Tree ID* that each packet contains. The *Tree ID* is the unique number generated for each packet by the OPNET simulator. For replicated packets, the *Tree ID* should be the same. A list is maintained to hold the *Tree IDs* already seen by the node so that subsequent packets received with the same *Tree IDs* are discarded without letting them reach the transport layer. A moving window of *Tree IDs* is maintained in this list to avoid the list becoming larger. But, a sufficient window size is maintained to allow for the *Tree ID* rollover to occur. In the case of packet replication, the receipt of a single packet results in the new changes making copies of the original packet and storing them in a list. Each of these packets is run through the **ip_rte_packet_arrival** function to perform standard IP processing of the IP layer.

3.4 Implementation of RDM Routing in Real Environments

The previous section described how the RDM protocol has been implemented in the OPNET simulator. The techniques used in the simulator are not always possible to be used in a real environment in the manner they are realized in the simulator. Therefore, this section highlights these areas and proposes solutions to implement the RDM protocol in a real environment.

3.4.1 Node Interference Computation

The computation of NI requires the knowledge of the IP traffic of other nodes in the neighborhood. The protocol stacks implemented in computers operate in a manner where only the IP packets that are destined to a node are sent up the protocol stack, disregarding the packets that are not destined to it. This is done by the link layer of a protocol stack. There are many tools in use today that require knowing about all the packets seen by the link layer. An example is a packet sniffer such as *Wireshark*. To cater to these requirements, most operating systems today have a special interface called the packet capture interface to the link layer to capture all the packets seen by the link layer. To enable this interface, the link layer that controls a certain network interface must be placed in a promiscuous mode. When the network interface is in promiscuous

mode, the packet capture interface receives an image of any packet seen by the network interface. Use of the promiscuous mode may result in taking more processing time to send unwanted packets form the MAC layer to the upper layers. Section 10.5 of Appendix II shows the results taken to compare the performance when setting the network interface to promiscuous mode enabled and disabled. These results do not show any significant difference in performance between two modes. By analyzing the address information present in all the packets received through enabling promiscuous mode, the two parameters of the NL and the NI can be computed as explained in section 3.1.2.

3.4.2 Avoiding loss of RREQs Messages

The computation of the INL as in section 3.1.2 is based on an assumption that there is no loss of RREQs during the propagation of RREQs in the network. In a real environment with IEEE 802.11 technologies, broadcast messages can easily be lost during the propagation, since there is no WLAN-ACK for the broadcast messages. Therefore, it is easy to lose RREQ messages. This causes a difficulty in completing the INL. For example, if a RREQ broadcasted by node 7 collides and is not received by node 16 in Figure 3-1, INL of node 16 does not have node 7 as an interfering node. The following two solutions are proposed in order to avoid/reduce the loss of RREQ messages.

Proposal 1: When forwarding a RREQ, each intermediate node should send a RREQ while introducing a jitter as explained in [53]. This requires assigning different values for jitter for the neighboring nodes. This makes sure that neighboring nodes do not send the RREQs at the same time, hence avoiding collisions. This proposal requires assigning/changing the values of jitter of the nodes dynamically in a real network.

Proposal 2: When an intermediate node forwards a RREQ it should hear the same copy that is forwarded by the other intermediate node in a multi-hop wireless network. In order to avoid creating routing loops, the same copy of the RREQ is not processed further as explained in section 3.1.1.1. However this information can be used to verify that the RREQ forwarded by a node has been received by another node. If an intermediate node does not receive the same copy within some time period, it can forward the original copy again. This requires an intermediate node to maintain a copy of a RREQ for some time. This solution does not detect RREQs lost in the last hop with D, since D does not broadcast the RREQs any further.

The above two proposals have already been implemented and are working successfully in the RDM aware DYMO protocol in the OPNET simulator. Table 3-5 compares the INLs computed for two grid topologies both in the simulation and theoretically. The simulation results are taken for two cases.

- Case 1: Each node is configured to broadcast a RREQ message with a varying jitter. The jitter of each node is defined by the uniform distribution to vary between 1ms and 3ms. Further, some of the selected nodes are used to enable the implementation of the proposal 2 due to the overhead that it introduces. For example, in the 3x3 grid topology, the middle node keeps the copies of RREQs broadcast and if it does not hear back the same RREQ during a predefined

period, it starts rebroadcasting the same copy. The middle node is selected, since it has a higher probability of loosing packets due to collisions from surrounding transmissions.

- Case 2: Each node transmits RREQ messages as specified in the standard protocol (neither using proposal 1 nor proposal 2).

Table 3-5 Computation of INL in the Simulation

Scenarios	Total no: of nodes in all INL			% of completion of INL w.r.t. theoretical computation	
	Theoretical	Case 1	Case 2	Case 1	Case 2
3x3 Grid (9 nodes)	20	20	16	100%	80%
5x5 Grid (25 nodes)	76	74	64	97.36%	84.21%

Table 3-5 shows that the use of proposal 1 and 2 increases the probability of completely computing the INL. In case 2, most of the RREQs are lost at the destination in 3x3 grid topology, since the probability is higher that two RREQs are propagated towards the same destination at the same time. In contrast, this will not happen in the 5x5 grid topology due to the existing BTL. But, most of the RREQs are lost in 5x5 grid topology, where nodes are carrying the BTL.

3.4.3 Implementation of Distribution Methods

Implementation of flow distribution is based on the ability of an operating system to identify IP packets in terms of different criteria and subsequently direct them to the correct path. This behavior can be implemented on a Linux platform using the IPTABLES and the IPROUTE2 functions [54]. IPTABLES is used to mark packets based on different criteria as follows [15, 55, 56]:

1. based on contents of the DSCP field for IPv4 or values of traffic class and flow label in IPv6
2. type of protocol (e.g. TCP, UDP)
3. source port/destination port
4. ranges of source ports/destination ports

IPROUTE2 is used to route these marked packets to the respective routing path. Splitting of packets is done using a feature available in IPROUTE2. The IPROUTE2 function enables the distribution of packets in a round robin manner with different weight factors.

Replicating packets (at the sender) and discarding redundant packets (at the destination) have to be done at the IP layer. Although the replicating and discarding has been implemented in the OPNET simulator using the unique ID used for each packet in the simulator, the following proposals therefore, discuss how a packet can be identified uniquely at the IP layer (both TCP and UDP flows) in a real implementation of the RDM protocol when replicating.

Proposal 1: Attach a unique ID to each outgoing packet in the transmitter's IP layer. This ID can be inserted into a packet's optional header (Options Field in IPv4 packet and Option Header in IPv6 packet). However, this method introduces an additional overhead for each data packet.

Proposal 2: There is a possibility of exploiting the "checksum" field in the transport headers to check the uniqueness of a packet. The checksum field is computed with the IP header, transport header and application data. However, two packets might share the same checksum if the application or control data they carry is identical. An advantage of this method is that it does not generate any additional overhead to the communication.

Proposal 3: The two proposals above can also be combined. More specifically, proposal 1 can be taken to insert a field with a short range (e.g. 1 byte) for a new ID variable. Meanwhile, the checksum can also be used to identify a packet's uniqueness together with the newly introduced ID. That way, the packets which carry the same application data are distinguished.

Table 3-6 Functions of RDM protocol

Functions of RDM	Implemented features in the OPNET Simulator	Feasibility of a real implementation
RDM Route Discovery Process		
A. Detection of Routing loops	Using node's IP address & accumulating path details	Easy implementation with the protocol carrying path the details (e.g. DSR, DYMO)
B. Avoidance of unnecessary flooding of RREQs	As explained in 3-2 (k is set to 1)	As explained in 3-2
RDM Path Selection Criteria		
A. Computation of the BTL	By computing the NL at the link layer	NL can be computed by setting the wireless interface to promiscuous mode
B. Computation of the Mutual Interference	Using RREQ messages	Both solutions explained in section 3.1.2 are possible
RDM Flow Distribution Methods		
A. MF Distribution	Yes (identifying the flow with ToS field)	Possible using IPTABLE & IPROUTE2
B. SF – Splitting	Round robin distribution with different weights	Possible using IPROUTE2
C. SF – Replicating	Yes – by discarding redundant packets using an ID	3 possibilities are discussed (section 3.3.3.3)
RDM Path Maintenance	As in standard DYMO protocol	Possible with the transmission of data or other control messages
RDM Dynamic Path Evaluation	No	Possible, but with an optimized way to reduce additional control messages

3.5 Conclusion

This chapter details the RDM protocol and the implementation. It focuses on the functions of the RDM protocol when discovering a pair of paths with the least interference and the least congestion. It also gives an overview on how to maintain and utilize the discovered RDM paths when distributing traffic flows. The second section explains how these functions are implemented in a simulated environment. The RDM protocol has been implemented in the OPNET simulator by extending the reactive protocol called DYMO. This chapter proposes how some of the features of the RDM protocol can be implemented in a real implementation. Table 3-6 details the summary of the RDM functions explained in this chapter.

4. Review of Application Performance over Wireless Multi-hop Ad hoc Networks

Today's Internet based applications mainly use TCP and UDP protocols as the transport layer protocol. These protocols have been designed mainly considering wired networks. These protocols behave differently in wireless networks, especially in multi-hop ad hoc wireless networks. The first section details how TCP reacts in multi-hop ad hoc networks discussing the results of previous research carried out in the area of improving TCP performance. It further details the TCP behavior in multipath routing and the performance of TCP when UDP applications are present. The second section gives a brief outlook to the research work analyzing the performance of UDP based applications. The last section gives an overview on the application performance analysis done in this thesis, highlighting the differences compared to previous analysis.

4.1 TCP behavior in Wireless Multi-hop Networks

TCP guarantees delivery of packets to the application layer in the order which they are sent by the sender. TCP is the most widely used protocol for applications such as WWW browsing, file transfer and mail transfers.

4.1.1 Overview of TCP Basics

TCP transfers data in the form of segments (TCP-Data). The application data can be segmented to either one or multiple segments depending on the size of data generated by the application. TCP uses acknowledgements (TCP-ACK) to maintain the correct order of delivery of packets. The TCP receiver only acknowledges successfully received, in-sequence data. Consequently, when receiving out of sequence data, TCP responds with duplicate acknowledgements, while keeping out of sequence data in the *out of order list*. The *out of order list* is processed as soon as the TCP receiver receives missing segments. TCP adapts a window based flow control mechanism, changing window size according to the network conditions. This allows sending new data when old data is acknowledged. The TCP sender maintains its congestion window (CWND) based on the feedback from the network.

At the start of the connection setup process, the TCP sender sets the Slow Start threshold (SS-threshold) to the size of receiver buffer (or advertised window) and initial CWND to the size of two segments. TCP sender tries to estimate the initial network capacity by doubling the CWND for each successful receipt of TCP-ACK [57, 58]. This phase is called the Slow Start (SS) phase. TCP stops increasing the CWND when a loss

of a TCP-Data segment is detected. The detection of a loss is identified either by the receipt of three duplicate acknowledgements (DupACKs) or by the expiry of a retransmission timer. After detecting a lost segment, TCP sets SS-threshold to half of its previous value and continues in Congestion Avoidance (CA) phase. TCP increases the CWND by one segment for each successful receipt of TCP-ACK during CA phase.

The TCP sender computes a retransmission timer (called RTO) for each TCP-Data segment before transmitting. The RTO is computed based on the previously measured RTT (Round Trip Time). If a corresponding TCP-ACK is not received before the RTO expires, the corresponding TCP-Data segment is considered to be lost and it is retransmitted and RTO is doubled. When a timeout happens, TCP assumes network congestion and starts from the SS phase and sets SS-threshold to the half of the CWND before the packet loss. TCP also uses *fast retransmit* [58] to initiate retransmission sooner. If the TCP sender receives three consecutive DupACKs, it initiates retransmission without waiting for the expiry of RTO. *Fast retransmit* is followed by *fast recovery* algorithm. During *fast recovery*, TCP retransmits the unacknowledged TCP-Data segment, reduces the SS-threshold to half of current CWND, and CWND is set to the new SS-threshold value plus the number of DupACKs received.

There exist different TCP versions that vary in the way how packet losses are processed. The TCP Reno version which is widely used contains the features of fast retransmit and the fast recovery algorithms. The TCP Reno version is used in all the examples discussed in this chapter.

4.1.2 TCP Performance in Multi-hop Ad hoc Networks

TCP performs poorly due to the TCP sender's inability to determine the cause of a packet loss properly in wireless multi-hop ad hoc networks. Since TCP is designed for wired networks, the TCP sender assumes that all packet losses are caused by congestion. The channel access delay varies in 802.11 based multi-hop ad hoc networks. Channel access delays are dependent not only on traffic congestion but also on interference from neighbors in the vicinity and possibly nodes further away. Therefore, the varying delay in the link layer can trigger a change of TCP parameters as explained below.

The effect of TCP performance in wireless multi-hop ad hoc networks are discussed in [59-61] in detail. In summary, TCP performs negatively in wireless multi-hop ad hoc networks mainly due to the following characteristics of wireless networks, even in the absence of congestion in the network.

- Sudden packet losses that occur in wireless ad hoc networks
- Route breaks due to node mobility
- Unfairness of channel access

4.1.2.1 TCP Reaction to Sudden Packet Losses

A sudden packet loss can occur in wireless networks even without any congestion. This happens due to node mobility, collisions of data transmission, interference from other

nodes and so on. Then, the packets can be dropped by the link layer itself due to exceeding the maximum transmission attempts.

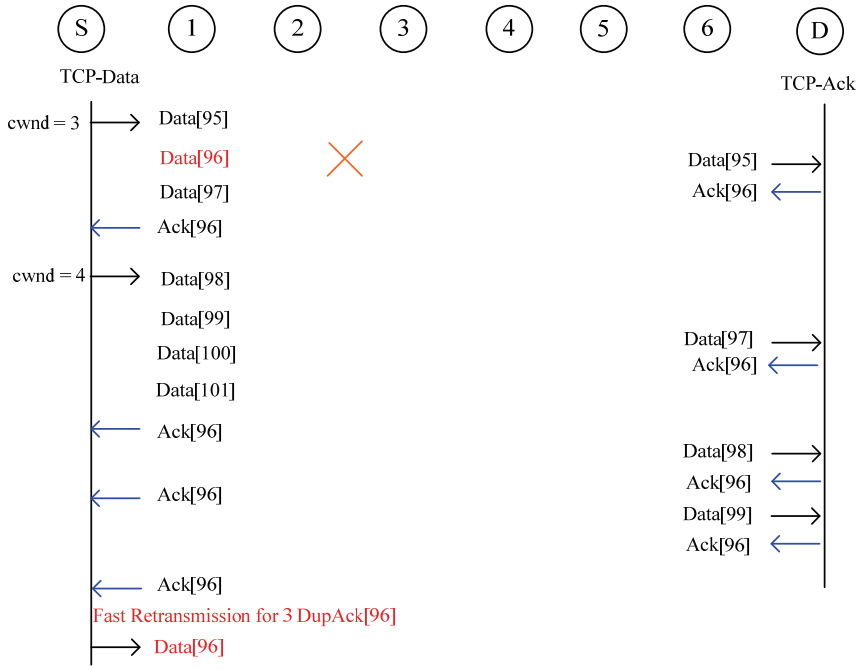


Figure 4-1 TCP reaction to a sudden packet loss

Figure 4-1 shows a TCP reaction to a sudden packet loss in a multi-hop ad hoc network consisting of 8 nodes. In this network, each node can transmit data only to its neighboring nodes and the node S initiates transmitting TCP-Data to the node D. It is assumed that only one hop neighboring nodes are within each other's interference range for the simplicity of explanations.

When the TCP receiver (i.e. node D) receives a TCP-Data segment, it sends the TCP-ACK indicating the next TCP-Data segment it expects. For example, when the TCP sender (i.e. node S) receives TCP-ACK with the number 96, the TCP-sender knows that the TCP-Data up to 95th segment has been received in order. Then the TCP sender starts sending the next TCP-Data segments to be sent by increasing its CWND. Assuming a 96th TCP-Data segment has been dropped by the link layer on the way to the TCP receiver, and then the TCP receiver must send the TCP-ACK for the missing segment 96 for the receipt of each subsequent segment. When the TCP sender receives the same copy of the TCP-ACKs (called DupACKs) more than 3 times, the TCP sender thinks that the packet is lost due to network congestion. Therefore, the TCP sender triggers fast retransmit and recovery algorithms by sending the 96th TCP-Data segment again. If the TCP sender does not receive 3 DupACKs before the expiry of the RTO for the 96th

segment, the TCP sender retransmit the 96th packet by reducing the CWND to 1 TCP-Data segment and starts with the SS phase. This causes the TCP throughput to reduce drastically. As shown in Figure 4-1, the TCP sender might not receive enough DupACKs before the expiry of the RTO of the 96th segment due to two reasons, i.e. the TCP receiver might not receive subsequent segments after the lost segment or the TCP-ACK may be dropped during the transmission via multiple hops.

4.1.2.2 TCP Reaction to Node Mobility

The route failures can occur in mobile wireless networks due to the change of network topology triggering a route discovery. There is no possibility of delivering TCP-Data or TCP-ACKs during a route re-discovery. Therefore, TCP timeouts may occur. A longer route discovery time has a negative impact on the standard TCP congestion control mechanism. In the standard TCP protocol, when a retransmission timeout happens, TCP sender retransmits the lost packet and doubles the RTO. This procedure is repeated until the lost packet is acknowledged. An exponential back-off of the RTO helps TCP react to congestion smoothly. But, in case of a route failure in wireless ad hoc networks, TCP tends to increase the RTO rapidly even when there is no congestion. The standard TCP assumes that the RTT can be measured accurately assuming that links are stable and the capacities of the links are fixed. These assumptions are not valid for wireless multi-hop ad hoc networks.

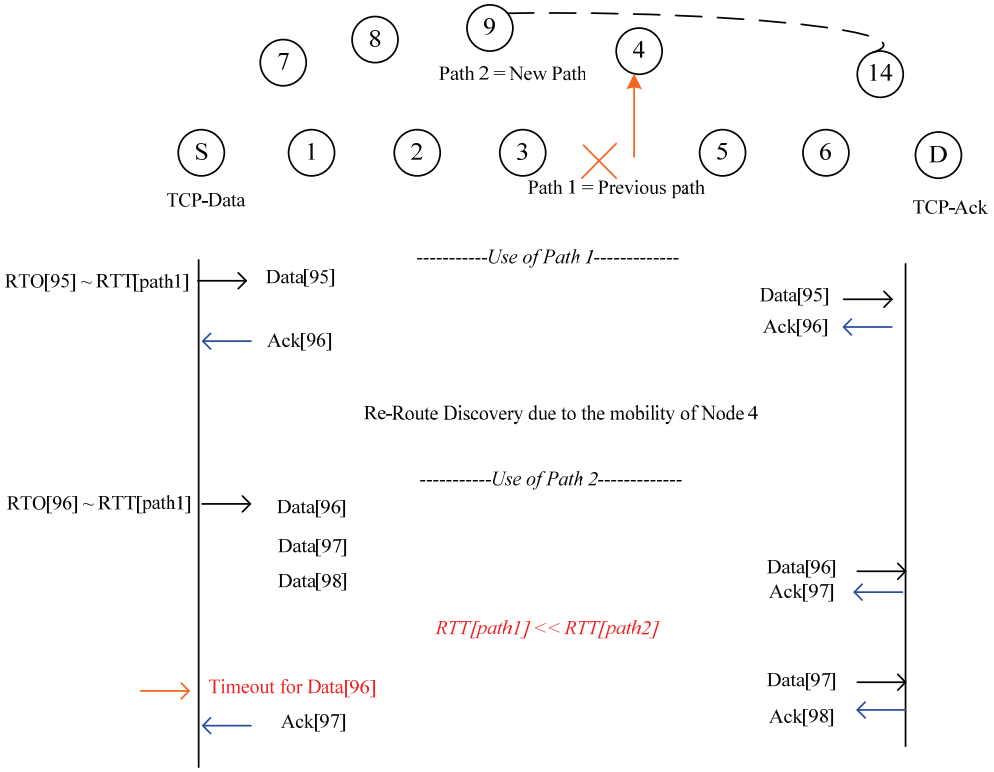


Figure 4-2 Premature TCP timeout after a new route re-discovery

Furthermore, the computation of the RTO does not match with the newly discovered path, after a new route discovery. In this case, TCP timeouts could occur, if the RTT of the newly discovered path is much higher than the previously used path.

As shown in Figure 4-2, path 1 is broken and path 2 which has a higher RTT than path 1, is discovered. When the TCP sender sends the first TCP-Data over path 2, it computes the RTO for the 96th segment based on the RTT of path 1. The computed RTO may be lower than the actual time that a corresponding TCP-ACK (Ack[97]) takes to reach the TCP sender via the newly discovered path. In this case, the TCP senders RTO expires even before the receipt of the TCP-ACK for the 96th segment.

4.1.2.3 Triggering of Route Failure at the Network Layer

In IEEE 802.11 based networks, RTS/CTS messages are used to avoid the hidden node problem. The use of RTS/CTS introduces additional delay in the transmission, though the number of packets dropped can be reduced with RTS/CTS messages. For example, consider the transmission of an RTS from the node 1 to node 2 in Figure 4-1. Node 2 may receive the RTS packet correctly but may be unable to send the corresponding CTS back to node 1. This happens, for example, if node 3 is sending data to node 4. During the transmission between nodes 3 and 4, node 2 reserves the channel by sending CTS to node 3. Failing to receive a CTS packet from node 2 after the specified number of retransmissions, node 1 quits and drops the data packet. Node 1 might not get a chance to send any data to node 2 due to the transmission between node 3 and node 4. At this point, a route failure at the network layer can be triggered by node 2 due to the unavailability of data transmission between node 1 and node 2. No data can be sent until a route is discovered again by the network layer. If the route discovery process is not fast enough, the TCP sender may timeout initiating the SS phase.

4.1.2.4 TCP Reaction to the Capture Condition

The capture condition in wireless network occurs when some nodes completely capture the channel access for one direction, while preventing transmissions in the opposite direction. The capture effect causes the most active connections to dominate the shared channel. This happens due to the binary exponential back-off mechanism of 802.11 MAC protocol. Binary exponential back-off favors always the last successful node as discussed in examples given below. TCP throughput can be degraded due to the capture condition under following situations.

4.1.2.4.1 Capture Effect on Shorter and Longer TCP Connections

The length of the connection in terms of number of hops used is also a fact to be considered when determining the TCP throughput. The shorter TCP connections can more easily capture the channel than the longer TCP connections [62] as shown in Figure 4-3. The shorter TCP connection is established between node 3 and node 4, while the longer TCP connection is established between node S and node D over 7 hops. The TCP-Data and TCP-ACK propagate faster over shorter TCP connections

which have lower hop counts or least congested links. A lower RTT of a connection means successfully received TCP-Data can be acknowledged faster and the TCP CWND also grows rapidly. So the TCP-Data and TCP-ACK packets from the short connection pass the shared nodes more frequently, and the short connection has a better chance at capturing the shared medium. As shown in Figure 4-3, node 4 gets the chance to access the channel more frequently than the node 6. Therefore, node 6 has to drop 61st TCP-Data segment after 4 tries of unsuccessful RTS/CTS exchange. In this example, TCP-Data transmission from node 6 is successful after completion of the shorter communication between node 4 and node 3. The capture effect is more conspicuous when the difference between the lengths of two connections (i.e. variations of RTT) increases.

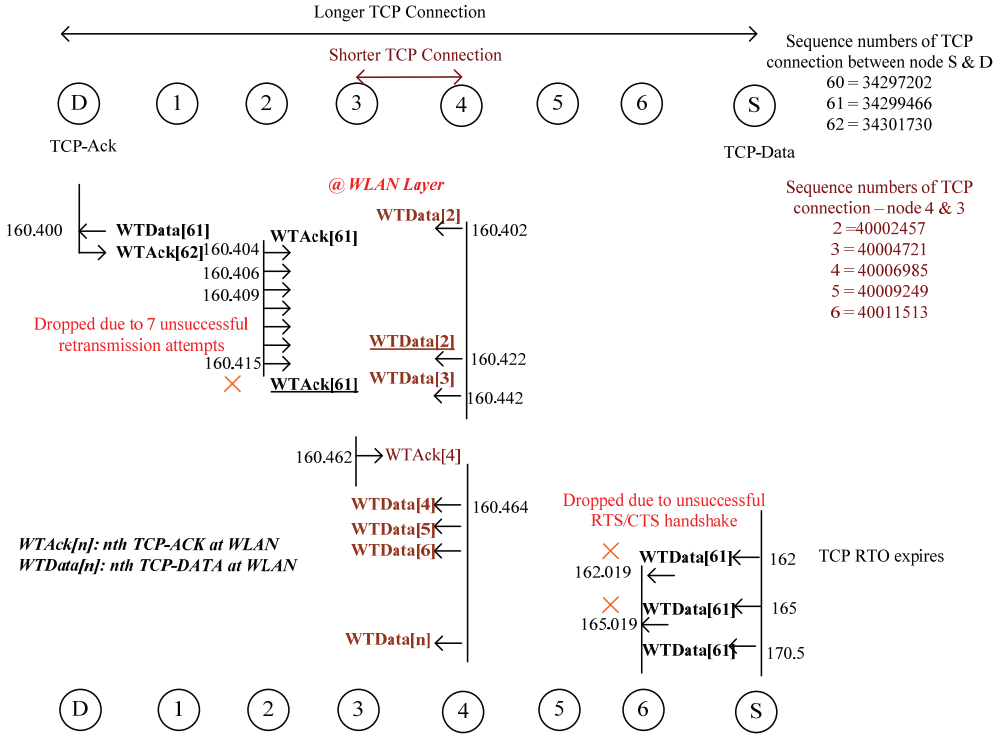


Figure 4-3 Shorter TCP connection (between node 4 and node 3) sends packets more frequently than the longer TCP connection (between node S and node D)

4.1.2.4.2 Capture Effect on Multiple TCP Connections

If there are multiple TCP connections sharing the same intermediate nodes, the connections which start earlier or more heavily loaded ones may have a higher probability of capturing the channel [62].

For example, if an earlier connection is in a CA phase and the later initiated connection is in an SS phase with a smaller congestion window, then the earlier connection has a high probability of capturing the channel at the sharing intermediate nodes. This happens since the earlier connection has more packets to be sent than the later

connection. In this situation, the earlier connection increases its CWND, while the later connection decreases CWND for each lost packet.

4.1.2.4.3 Conflict between TCP-Data and TCP-ACK

There might be a conflict of transmission between TCP-Data and TCP-ACK due to the capture condition. This is illustrated in Figure 4-4, using the previously used example network consisting of 8 nodes.

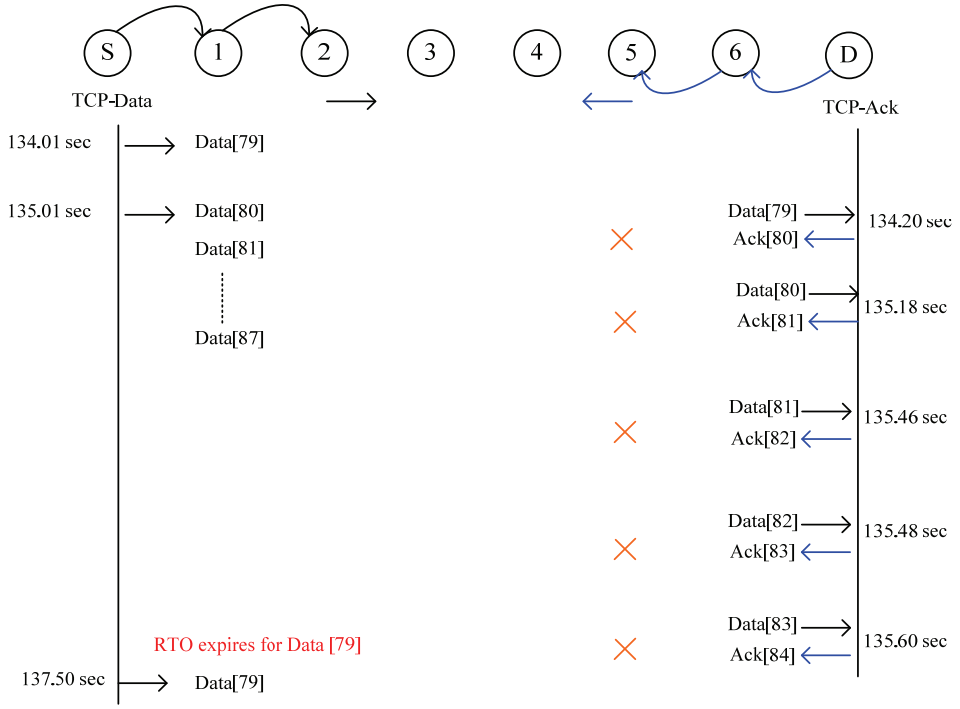


Figure 4-4 TCP-Data and TCP-ACK conflict due to the capture condition

Once the TCP connection has been continually used for a while, node S starts increasing its CWND and sending multiple TCP segments at once (from packet 80 to 87, as shown in Figure 4-4). For example, assume that the previous TCP-ACK for packet 79 is on the way to the TCP sender. This creates an asymmetric transmission in the network, by sending multiple larger packet sizes towards the TCP receiver (i.e. TCP-Data) and smaller size of TCP-ACKs towards the TCP sender. Assuming that RTS/CTS is enabled, for example at node 2, it always gets the channel to send multiple TCP-Data to node 3. Each time, node 3 is transmitting; node 5 does not receive any CTS packet from node 4 (node 4 has already heard the CTS from node 3 due to the RTS/CTS handshake done between node 2 and node 3). Therefore, node 5 has to drop sending the TCP-ACK to node 4 after several unsuccessful attempts to receive the CTS from node 4. The capture condition can easily occur for TCP data transmissions when sending multiple TCP-Data segments and loosing the TCP-ACK traveling in the opposite direction. The only chance that node 5 has to access node 4 is to send an RTS before node 2 sends an

RTS. Since node 2 has more data to send successfully, node 5 hardly wins the contention (since binary exponential back-off in IEEE 802.11 MAC favors always the successful node).

As shown in Figure 4-4, the TCP sender does not receive any acknowledgments though the TCP receiver sends multiple TCP-ACKs for each successfully received TCP-Data. Therefore, TCP sender's timeout occurs and it has to reduce the CWND initiating the SS phase.

4.1.2.5 Summary - TCP Reactions in Multi-hop Ad hoc Networks

In summary, TCP is not designed to distinguish between the packet losses due to sudden errors/link failures and delays due to congestion. Therefore TCP often reduces its CWND even without having network congestion in wireless multi-hop ad hoc networks and thus showing poor performance. The following characteristics of wireless multi-hop ad hoc networks result in the degradation of standard TCP performance.

- Random packet losses that occur on wireless links.
- Link failures at MAC layer (due to unfairness in channel access) can easily trigger route failures at the network layer. This increases the number of unnecessary route discoveries.
- The standard TCP retransmission timeout grows too fast during a route re-discovery process.
- After a route re-discovery, the newly discovered routing path can have a higher RTT and TCP could trigger timeouts.
- TCP traffic is bursty in nature causing unfairness in accessing the channel. The 802.11 MAC protocol induces unfairness between multiple TCP connections, depending on the number of hops and the load used. Even between the TCP-Data and TCP-ACK packets of the same connection, unfairness is created when sending multiple TCP-Data segments at once while smaller size of TCP-ACK tries to propagate along the opposite direction.

4.1.3 Improvement to TCP Performance in Multi-hop Ad Hoc Networks

There are a considerable number of proposals which discuss how to improve the TCP performance over wireless multi-hop ad hoc networks. These proposals fall mainly into two categories:

1. modifying standard TCP behavior defined for wired network
2. replacing or modifying the IEEE 802.11 MAC protocol

4.1.3.1 Modifications to Standard TCP

Most of the proposed work discusses how to change the TCP parameters or the standard TCP behavior to perform better with wireless multi-hop ad hoc networks. Most of the proposed modifications are related to how TCP can distinguish between packet losses due to mobility and congestion at the 802.11 MAC layer. Once TCP detects the link failure due to route breaks, it freezes the TCP state (CWND size and RTO interval) until a new path is established [63].

The intermediate nodes can detect the link/route failures and send feedback to the TCP sender to inform about the network conditions. This feedback can be used by the TCP sender to avoid reacting to the failures as if congestion has occurred. Previous work of TCP-Feedback (TCP-F) [63] and Explicit Link Failure Notification (ELFN) [59] discuss how an intermediate node can send a feedback message back to the TCP sender. In TCP-F, it explicitly sends a Route Failure Notification (RFN) to the TCP sender, when an intermediate node detects a route failure. In ELFN, it uses probing packets periodically to detect the failures.

In [64], a mechanism called fixed-RTO is proposed to avoid unnecessary RTO back-off during the route breaks. Consecutive TCP timeouts are considered to occur, mainly due to the route breaks. Therefore, after retransmitting the lost packet, a fixed-RTO is maintained until the route is reestablished.

In the ENhanced Inter-layer Communication and control (ENIC) proposal [65], a new retransmission timeout value is calculated after a route change, in a heuristic fashion. The RTO is computed based on the hop counts of the previous and the newly discovered routing paths.

In Ad hoc TCP (A-TCP) [66], a thin layer is inserted between TCP and IP in order to avoid changes in the standard TCP/IP suite. A-TCP listens to the network state information provided by ECN (Explicit Congestion Notification) messages and by ICMP “destination unreachable” messages and then puts the TCP sender into an appropriate state. The congestion control of standard TCP is modified to improve its throughput by modifying congestion control and retransmission algorithms.

In [61, 62], the authors show that the limitation of the maximum CWND (e.g. to 4) to a smaller value helps TCP to perform better in IEEE 802.11 multi-hop ad hoc networks. Smaller CWND makes it possible to avoid capturing the channel to send data only in one direction and also avoiding the unfairness of channel access. This also helps to avoid triggering of unnecessary route failures at the network layer (see section 4.1.2.3).

4.1.3.2 Modification to IEEE 802.11 MAC

As explained above, the standard TCP does not work well in wireless networks compared to wired networks. The IEEE 802.11 MAC protocol is the most widely used MAC protocol in wireless multi-hop ad hoc networks. This section gives a brief overview of previous research in the area of improving TCP performance by modifying the 802.11 MAC protocol.

The IEEE 802.11 MAC protocol is designed to provide reliable data exchange over a shared medium. It employs link layer acknowledgements to guarantee point to point (node to node) reliability. In contrast to 802.11 MAC, TCP guarantees end to end reliability. TCP uses a back-off when it encounters any packet loss while 802.11 MAC uses back-off to avoid conflicts when accessing the shared channel.

Previous work of [61] shows how TCP performs in different MAC protocols like CSMA, MACA (Multiple Access Collision Avoidance) and MACAW (Multiple Access Collision Avoidance for WLAN). This proves that CSMA protocols can be improved with mild back-off and selective scheduling to perform better when used with TCP applications.

In [62], it is shown that intelligently tuning the parameters used in both TCP and the IEEE 802.11 protocols improves the TCP performance. These results show that TCP degrades mainly due to link failures at the MAC layer, when having an aggressive TCP-Data transmission. This increases the contention at the MAC layer, which triggers the route failures at the network layer. This paper proves that better TCP performance can be achieved by reducing the CWND to a smaller value. It further shows that an increase of the retransmission limit at the MAC layer also improves the TCP performance in stationary scenarios.

Significant work has been done at developing novel MAC layer schemes to enhance the application performance. In [67], the authors propose a hybrid scheme where the senders as well as the receivers are allowed to initiate the collision avoidance handshake. This protocol is compatible with the IEEE 802.11 MAC protocol. It has an additional feature which is a simple queue management technique to handle the capture condition. Protocols other than the IEEE 802.11 MAC have also been proposed [68, 69] to be used in multi-hop ad hoc networks. These protocols are not explained in detail here since the improvements to the MAC layer is not within the scope of this thesis.

4.1.4 TCP Performance on Multipath Routing

This section highlights the effects of TCP performance when using multipath routing. The performance of TCP is affected for the same reasons that are explained in section 4.1, when using multipath routing in multi-hop ad hoc networks. As mentioned in section 2.2.2, multipath routing can be used to distribute TCP flows among multiple routes simultaneously (SUM) or using one route at a time (APR), keeping the others as backup paths.

4.1.4.1 TCP Performance over APR

TCP applications are used to send over a single path, when using APR. The use of APR reduces the number of route discoveries in mobile ad hoc networks since it can select a path from the backup paths in case of a route failure. Therefore, TCP should perform better compared to SP since issues such as RTO expiries during route discoveries are alleviated. TCP performance when using APR is analyzed in [29, 30, 70] in detail showing that the performance can be improved with APR in mobile environments.

4.1.4.2 TCP Performance over SUM Routing

In contrast to APR, SUM can affect the TCP performance negatively if the utilized paths are interfering with each other and receiving more out of order packets due to

splitting a single TCP connection among different paths with different properties. Using SUM to split packets of a single TCP connection may lead to the following problems.

- *Average RTT estimation is not accurate*: The average RTT computed by multiple paths may be much shorter than the RTT in the path with the longest delay. Therefore, the TCP sender may prematurely timeout for the packets which are sent over the path with the longest delay.
- *Out of order packets*: Packets going through different paths may arrive at the destination at different times causing packets to be out of order. This can trigger fast retransmit algorithm thereby reducing TCP throughput (see Figure 4-5).

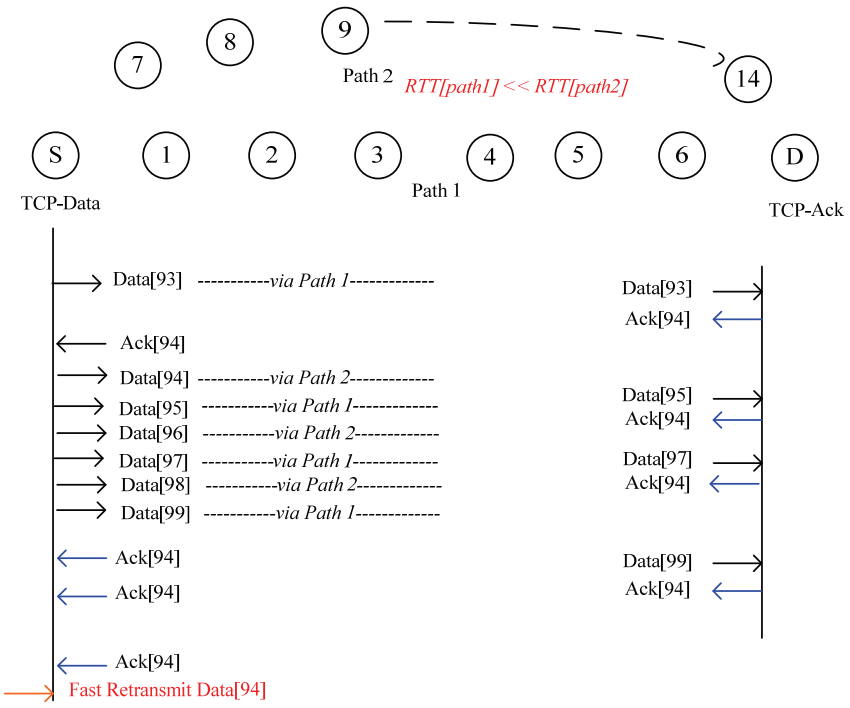


Figure 4-5 Out of order TCP-Data delivery with SUM routing

A detailed investigation of TCP performance over multipath routing based on DSR is discussed in [71]. TCP Reno is used with a fixed RTO scheme. It is observed that in most of the cases considered, the use of multiple paths simultaneously may actually degrade the TCP performance. This is in contrast to UDP traffic, which usually gets uniformly good performance over multipath routing, TCP performance over multipath routing shows negative results in all the investigated scenarios. This is mainly due to the TCP traffic reacting to RTT and other network parameters very sensitively (as shown in Figure 4-5). This proposal uses the maximally disjoint paths as multipath routing. Those routes could introduce mutual interference between the paths causing poor performance for TCP. Further, the distribution of packets is not done based on any property of a path

such as RTT or the available bandwidth. This causes a higher number of out of order packet deliveries as shown in Figure 4-5.

The effects of multipath routing on TCP throughput (number of sequenced bits that a TCP receiver receives per second) in wireless ad hoc networks is evaluated in [29], considering multiple edge disjoint paths and multiple node disjoint paths. Multiple paths are used simultaneously with a scheduling policy, wherein, the number of packets that are distributed on a path is inversely proportional to the average RTT experienced on that path. This sends a higher number of packets on the least congested paths. In order to avoid TCP degradation during a re-route discovery, the TCP sender disables its retransmission timer and enters into a standby mode upon receiving a RERR packet. Results show that long TCP connections (split over paths with longer hops) benefit to a certain extent while short TCP connections may even suffer a slight degradation in throughput. This paper shows the improvement for simultaneous use of multipath routing when splitting packets of a TCP connection over longer hops, in contrast to proposal 1. This is due to the attempt to bypass congestion by scheduling a larger number of packets on the paths that experience lower *RTTs*. Thus, the benefits are primarily due to the alleviation of the effects of link failures due to the congestion.

This proposal does not consider using paths simultaneously which have the least mutual interference. Simultaneous use of even node disjoints paths lead to poor performance if they are interfering with each other. In wireless multi-hop ad hoc networks, the RTT can vary due to many reasons. Therefore, the RTT measured during the route discovery or the RTT measured based on previous data is not the best parameter to predict the delay of a wireless link.

4.1.5 Multipath TCP

As explained in section 4.1.4, the receipt of out of order packets and wrong estimation of average RTT are the main problems associated with splitting of TCP packets into multiple routes with differing properties.

The latest proposal for TCP to perform better over multipath routing is called MultiPath TCP (MPTCP), which is designed for any kind of network (wired/wireless) with multipath routing. The standard for MPTCP is still being discussed at the IETF [72]. MPTCP allows a single TCP connection to distribute packets into multiple routes by maintaining multiple sub flows simultaneously. Each sub flow in MPTCP has its own congestion window so that MPTCP can continuously measure the congestion level on each sub flow and dynamically distributes packets considering less congested paths[73].

MPTCP is aware of the existence of multiple routing paths and creates a different sub flow for each path. MPTCP characterizes each path through the congestion experienced and it distributes the load among the paths based on the congestion. Each sub flow behaves as an independent TCP connection, with its own congestion control mechanisms.

This thesis focuses on the discovery of optimal multiple routing paths in wireless multi-hop ad hoc networks, without modifying any application or transport layers. However, MPTCP should perform better with RDM routing proposed in this thesis. MPTCP is not used to evaluate the performance together with RDM routes due to the lack of available implementations for the simulation environment used in this thesis.

4.2 UDP Performance in Wireless Multi-hop Networks

UDP (User Datagram Protocol) [74] is a connectionless protocol that provides a direct way to send and receive data over an IP network. It does not implement congestion control or congestion avoidance schemes like TCP and does not provide reliable end-to-end delivery. Therefore the issues that are discussed in section 4.1, are not severely affecting the performance of UDP based applications. The congestion in the network and the packet losses may result in higher delays, jitter and number of packets lost by the applications.

Most of the previous research has used UDP based applications to evaluate the performance of the standard ad hoc protocols [17, 50, 75]. Multipath routing protocols have used mostly CBR based UDP traffic, as discussed in section 2.3. These papers [27, 30, 36-38, 49] discuss how UDP performs in terms of data throughput, packet losses, average end-to-end delays and the number of route rediscoveries.

4.2.1 TCP Performance in the Presence of UDP

UDP applications utilize more bandwidth since UDP does not use any congestion control like TCP. Therefore, a UDP based applications can build up large queues on nodes (even at intermediate nodes). This queue build-up, and the subsequent medium capture, creates congestion in the routing path as well as neighboring paths. In such a situation, the TCP flows that intersect (or interfere) with the heavy UDP flows suffer delays and packet losses, causing poor performance. In [76], it is shown that the use of fair queuing improves TCP performance in such a scenario. It proposes how to improve fair queuing in multi-hop ad hoc networks by a method called back-pressure.

4.3 TCP/UDP Performance over RDM Routing

The above mentioned proposals mostly focus on improving the TCP performance by modifying the standard TCP and MAC protocols. Previous proposals on SUM analyze only splitting packets of a single TCP connection (section 4.1.4). In contrast to previous work, in this thesis, how to improve the TCP and UDP performance by using the least interfering and least congested paths simultaneously is analyzed in detail. The main contributions of this thesis, compared to the previous application performance analysis are listed below.

- Real applications are used instead of generating pure UDP or TCP based traffic. The applications used are:
    - UDP : bi-directional VoIP applications, bi-directional video transmission

  - TCP : FTP, HTTP (web browsing)
- The standard TCP/IP suite is used without any modifications
  - Standard TCP and UDP protocols
  - Standard IEEE 802.11b WLAN protocol
- Application flows are distributed based on 3 methods among simultaneously used RDM routes
  - *MF distribution*: a detailed analysis of MF distribution in multi-hop ad hoc networks with multipath routing has not been investigated earlier. This is analyzed in Chapter 5, by using both UDP and TCP based applications simultaneously
  - *SF distribution-Splitting*: only a few research focuses on splitting packets of a single flow for both UDP and TCP applications [27, 29, 70]. But, proper methods of distribution of packets are not discussed except in [29]. It discusses the distribution of packets based on the RTT measured using previous data transmissions
  - *SF distribution-Replicating*: the effect of replicating data has not been investigated in previous work. A detailed analysis of replicating packets among RDM routes is discussed in Chapter 6.
- Distribution criteria: This thesis introduces a novel mechanism to distribute multiple flows and packets of a single flow based on the PL, which is computed considering the BTL and the mutual interference between paths (section 5.2)
- Previous work analyses the performance of applications with SUM routing when selecting routes without considering the effects of mutual interference. This thesis considers the following combinations of routes to analyze the application performance.
  - Routes that are not interfering and least congested (FRDM routes)
  - Routes that are interfering and highly congested (NRDM routes)

5. Performance Evaluation of RDM Routing: SF and MF Distributions

This chapter is devoted to discussing the simulation results. The evaluation of results is done considering the non-interfering RDM routing, the interfering RDM routing and Single Path (SP) routing. When using RDM routes, two distribution methods, viz., single flow and multiple flow distribution methods are considered. The first section details the simulation environment, the scenarios and the applications used to evaluate the performance of the RDM routes. The second section is devoted to explaining the MF and SF distribution algorithms. The last two sections compare the performance of applications when using the SP and RDM routing together with the MF and SF distribution algorithms.

5.1 Simulation Environment & Scenarios

The RDM protocol has been implemented in the OPNET simulator as explained in section 3.3. Table 5-6 shows the parameters used in the simulation environment.

Table 5-1 Parameters of the simulation environment

Link Properties	IEEE 802.11b (ad hoc mode)
	PHY mode is set to 1 Mbps
	RTS/CTS enabled (RTS threshold = 80 bytes, see section 10.4)
	Tx power 100 mW
	Rx threshold -76dBm
	Large packets are fragmented (threshold is set to 2304 bytes)
	Buffer size 1024 Kbytes
	Retransmission retry if RTS/CTS is not enabled = 7
	Retransmission retry if RTS/CTS is enabled = 4
Propagation model	Free Space (used in OPNET Simulator)
Communication Ranges	Transmission Range – 600m
	Carrier Sensing Range – 600m
RDM Routing Protocol	Maximum number of RDM routes = 2
	Method of distribution – both splitting and MF distribution
	RDM routes are selected based on PL (see section 5.2)
	Node mobility – Random Waypoint Model

SP Routing Protocol	A route with lowest hop count is selected
Simulation Details	10 seeds (results are shown with 95% confidence interval)
	Simulation duration 1000 sec
Traffic Generators	Applications – both real time and bursty traffic (see section 5.1.2)
	BTL – CBR traffic generated with UDP

The following terms are used in the subsequent sections.

- *SF Distribution:* This refers to the splitting of packets of a single flow. It could be an audio, a video, an FTP download or a HTTP application. The distribution rate is defined based on the path properties as explained in section 5.2.2.
- *MF Distribution:* This is used when the sender (either node n_S or n_D) originates more than one flow at the same time or consecutively during the simulation period. When using RDM paths, the MF distribution sends packets from the same flow over the same path. How flows are distributed is decided based on the required bandwidth of active flows and the remaining bandwidth of a path as explained in section 5.2.1.
- *SP:* SP scenarios use only one path to send all available flows. This path is selected only considering the hop counts, i.e. the path with the lowest hop count is used as the SP in all the scenarios. Section 5.3.6 compares the standard SP with the lowest hop count together with the SP discovered by the RDM protocol. The RDM protocol selects a path with the least interference and least BTL as the SP.
- *RDM:* RDM scenarios (FRDM or NRDM), 2 node disjoint paths are selected if available considering the mutual interference and the BTL of the paths. When using RDM paths (FRDM or NRDM), the MF distribution is enabled, only if multiple flows are active. The SF distribution can be used in the presence of either a single flow or multiple flows.

5.1.1 Simulation Scenarios

The following scenarios are used to evaluate the performance of RDM routing using different applications.

5.1.1.1 Basic Topology

Figure 5-1 shows the use of SP and different types of RDM routes in a simple network topology, where S denotes the sender, n_S and D denotes the destination node, n_D. Broken lines in the NRDM scenario show the interfering links. This scenario is used to highlight the performance of RDM routes comparing to the SP and the NRDM routes. FRDM and NRDM scenarios use identical routing paths simultaneously. The paths are identical in terms of hop counts and also the level of mutual interference between paths.

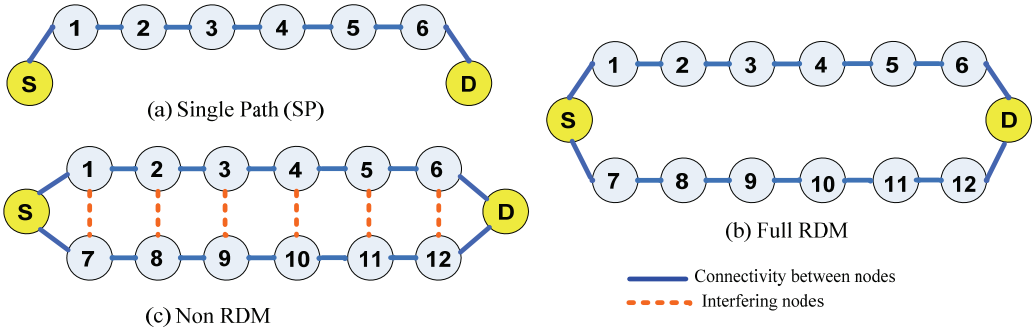


Figure 5-1 Basic topologies: SP, Full RDM and Non RDM routes

5.1.1.2 String Topology

The basic scenarios discussed above only consider the use of identical routing paths with identical hop counts and with an absence of any BTL. Figure 5-2 shows a 17 node network, where RDM paths have different characteristics. In this scenario, there exists a BTL between the node n_8 and the node n_9 of the shortest path. Without RDM routing, SP is chosen as the path with the lowest hop count, i.e. the middle path. RDM routing avoids the selection of this as the primary path due to the existing BTL and also due to the higher number of interfering nodes with the upper and the lower paths. It selects only the upper and the lower paths as FRDM paths.

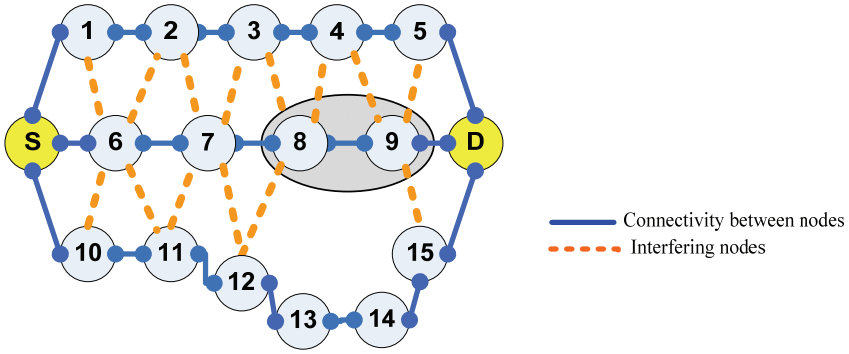


Figure 5-2: Possible alternate paths used in a 17 nodes topology. The BTL is configured between node 8 and node 9

As explained in section 3.1.2.2, the parameters used to select the primary path and the secondary path is given in Table 5-2 and Table 5-3. These parameters are calculated when the node n_8 is transmitting 4100 bytes/sec to the node n_9 and vice versa. It considers the lower path as the primary path due to this path having the least number of interfering links and least effect from the BTL (see Table 5-2). In this scenario, the upper path is selected as the secondary RDM path (see Table 5-3).

This scenario is used to evaluate the performance when using non identical RDM routes simultaneously. As shown in Figure 5-2, the upper and the lower paths have different hop counts and also a varying effect from the BTL in the middle path. Further, this scenario is used to compare the performance by changing the amount of the BTL used in the SP. The BTL is generated using a CBR UDP flow by utilizing the link between the node n_8 and the node n_9 with 6.56% , 18.56% and 37.12% for the BTL (Table 5-4).

Table 5-2 Parameters used to select the primary path - String topology

Path Selection Parameters	Middle Path	Upper Path	Lower Path
Num: of interfering nodes	22	18	18
Maximum BTL, $T_{\max}$ (bytes/sec)	4100x2 (n_8 or n_9)	4100x2 (n_4)	4100x1 (n_{12} or n_{15})
Accumulated BTL, T_{acu} (bytes/sec)	8200x2 + 4100x2	8200x1+4100x2	4100x2
Num: of hop counts, HC	5	6	7
Mutual Interference, I_{1r} (w.r.t. lower path)	8	-	0

Table 5-3 Parameters used to select the secondary path - String topology

α	Middle Path, PL_2 $I_{12}=1 \& T_2=1$	Upper Path, PL_3 $I_{13}=0 \& T_3=1$
0.7	1	0.3
0.2	1	0.8

Table 5-4 BTL used - String topology

BTL used	Throughput @ WLAN	% of link usage w.r.t. 1 Mbps of PHY mode
BTL-1 = 2x(125 bytes)/(0.05 sec)	2x205 bytes [9]/ 0.05 sec (8200 bytes/sec)	6.56%
BTL-2 = 2x(500 bytes)/(0.05 sec)	2x580 bytes / 0.05 sec (23200 bytes/sec)	18.56%
BTL-3 = 2x(500 bytes)/(0.025 sec)	2x580 bytes / 0.025 sec (46400 bytes/sec)	37.12%

5.1.1.3 Grid Topology

This scenario consists of 25 nodes as discussed earlier in section 3.1 (Figure 3-1). The evaluated routing paths between the sender (n_S) and destination (n_D) are shown in Figure 7-14. A unidirectional BTL is added to links connected to the 6 nodes of n_6 & n_7, n_8 & n_9 and n_{10} & n_{11} (nodes in the shaded area of the Figure 7-14) as BTL. In

[9] App data + UDP header (8) + IP header (20) + MAC header (28) & PHY header (24)

this scenario, all lateral neighbors are within both the transmission and the interference ranges. The diagonal neighbors are not interfering with each other.

Table 5-5 shows the selection criteria for the paths as explained in section 3.1.2, when generating the BTL by sending 20 packets (125 bytes each) per second from n_6 to n_7, n_8 to n_9 and n_{10} to n_{11}. When using the RDM routing paths, P_1 is selected as the primary path (with zero BTL and having the least number of interfering nodes). Computation of mutual interference w.r.t. P_1 gives the highest mutual interference for path P_2 (NRDM routes) and zero mutual interference for paths P_3 and P_4 (FRDM routes). This scenario represents 3 different types of routing.

- SP – P_1
- NRDM – (P_1 and P_2)
- FRDM – (P_1 and P_3) or (P_1 and P_4)

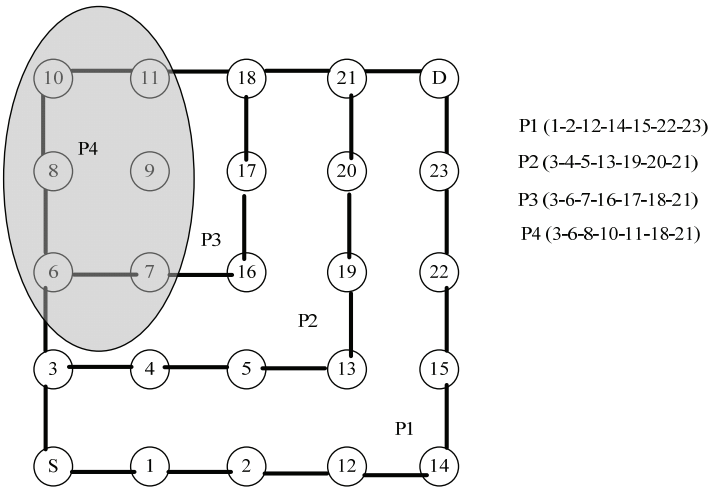


Figure 5-3: Possible routes in a 5x5 grid network with consideration of BTL. Nodes in the shaded area are configured with BTL

Table 5-5 Parameters used to select the primary path - Grid topology

Path Selection Parameters	P_1	P_2	P_3	P_4
No: of interfering nodes	20	26	24	20
Maximum BTL, T_{max} (bytes/sec)	0	4100x1 (n_3 or n_4)	4100x2 (n_6)	4100x3 (n_8)
Hop counts, HC	8	8	8	8
Mutual interference, I_{1r} (w.r.t. P_1)	-	6	0	0

From the above routing, P_3 is selected as the secondary RDM path since P_4 carries more BTL compared to P_3. In this scenario, the NRDM paths are selected when using lower values for the weight factor, α (see Table 5-9).

Comparing with the basic and the string topology scenarios, this scenario represents the use of RDM routes with non identical properties. Here, the secondary RDM path itself carries the BTL while the primary path is not carrying any BTL. This scenario is used to evaluate the performance of FRDM (P_1 and P_3) routes, comparing with SP (P_1) and NRDM (P_1 and P_2) routes.

Table 5-6 Parameters used to select the secondary path - Grid topology

α	PL_2 $I_{12}=1 \& T_2=0.33$	PL_3 $I_{13}=0 \& T_3=0.66$	PL_4 $I_{14}=0 \& T_4=1$
0.7	0.799	0.198	0.300
0.2	0.464	0.528	0.800

5.1.1.4 Random Topology

As shown in Figure 5-4, a random topology consisting of 30 nodes, distributed randomly in a 1.8 km x 2.4 km area, is also used to evaluate the performance of RDM paths. The "Source" node always sends application flows to the node "destination". This topology is used to take results for both SP and FRDM routes for the following two cases.

- Without any BTL. None of the nodes carry any BTL.
- With BTL. A higher BTL of 36% (denoted by BTL-3 in Table 5-4) is applied for the communications between the *node_15* and *node_16*.

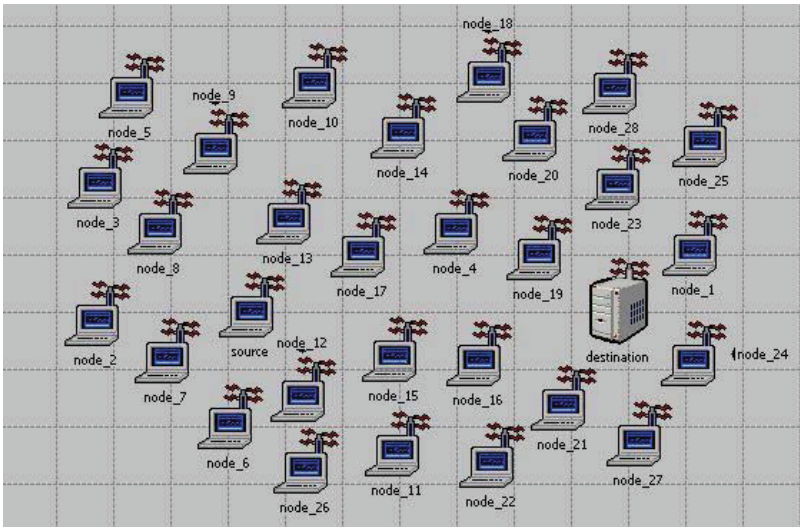


Figure 5-4: Random topology with 30 wireless nodes. The nodes other than the source and the destination are selected randomly as intermediate nodes for the discovered paths

5.1.1.5 Mobility Scenario

A mobility scenario is configured using a RWM model for a network consisting of 30 wireless nodes. The nodes are deployed randomly in a 1.8 km x 2.4 km area. The mobility is assigned to each node using the RWM model, which is configured to be used with constant speeds of 1 and 8 m/s by restricting the mobility area as shown in Figure 5-5. The following additional parameters are analyzed with the mobility to evaluate the performance of route discoveries.

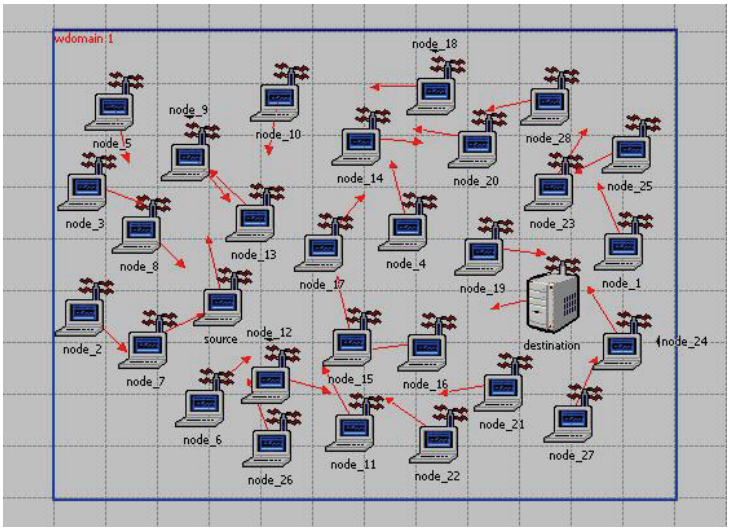


Figure 5-5: Mobility scenario with 30 wireless nodes. The nodes other than the source and the destination are selected randomly as intermediate nodes.

- *Number of RREQs generated by the originating node:* This shows the total number of RREQs generated either by the source or the destination node in order to discover a route.
- *Number of RERRs generated:* The RERRs are generated when any nodes detect a link failure to active neighboring nodes. This statistic shows the total number of RERRs generated by all the nodes in the network.
- *Number of successful route discoveries:* The originating node has to send more RREQs to find a route to the destination if the RREQs do not reach the destination or the RREPs do not reach the originating node due to mobility. Therefore, the number of RREQs generated is not always equal to the successful route discoveries. This statistic shows the total number of successfully completed route discoveries.
- *Route Discovery Time (RDT):* This shows the time taken to find a route. When using the RDM routing protocol, RDT is updated upon receiving the first RREP.
- *The percentage of usage of routes:* The RDM protocol discovers 2 disjoint routes (if available) and the next route discovery is not initiated until the expiry of both routing paths. Therefore, there is a possibility that the RDM protocol

also uses a single routing path if the other path has already been broken. When using one path, all the flows are forwarded via the non broken path. This statistic shows how many times the SP route is used and RDM paths are used simultaneously. It shows as a percentage w.r.t. the total simulation time.

5.1.1.6 Evaluation of proposed algorithms

This section compares the theoretical values against the simulated values of the main parameters computed during the RDM route discovery process. As discussed in Chapter 3, 2 parameters affect the selection of RDM routes: the computation of the BTL and the computation of the INL.

Computation of the BTL: This is implemented as proposed in section 3.1.2.1 using the weighted average. The weighted average is computed in every 4 second interval. Table 5-7 shows the comparison of the computation of the BTL for both theoretically computed and simulated values. The theoretical computation of the BTL is detailed in Table 5-2 and Table 5-5. Table 5-7 shows the computation of the BTL is done accurately in the simulation.

Table 5-7 Computation of maximum BTL

Scenario		Theoretical (bytes/sec)	Simulation (bytes/sec)
String Topology with BTL-1	Upper path	8200	8200
	Middle path	8200	8200
	Lower path	4100	4100
Grid Topology with BTL-1	P_1	0	0
	P_2	4100	4100
	P_3	8200	8200
	P_4	12300	12300

Table 5-8 Computation of INL based on the RREQ details

Scenario	Total number of nodes in all INLs		% of completion of all INLs
	Theoretical	Simulation	
Basic Topology – FRDM (14 nodes)	24	24	100%
String Topology with BTL-1 (17 nodes)	58	57	98.27%
String Topology with BTL-3 (17 nodes)	58	55	94.82%
5x5 Grid (25 nodes)	76	74	97.36%
Random Topology (30 nodes)	-	52	-

Computation of the INL: This is implemented as proposed in section 3.1.2.2 by evaluating the node details in each RREQ received. The INL is computed accurately only if all RREQs are propagated along the valid paths without any loss on the way. Table 5-8 compares the computation of the INL for both theoretically computed and the

simulated values. These results show that the INL is not computed accurately in the simulation when an increased BTL and a higher number of nodes are present. The probability of loosing RREQ messages becomes higher with the increase of the BTL and the number of nodes in the network. In the random topology, the theoretical computation of INLs is not shown due to the complexity of performing a manual computation of all possible paths.

5.1.2 Applications and Parameters used to Evaluate the Performance

The following applications are used to evaluate the use of RDM routes in different scenarios. They are real time applications that generate data traffic using the UDP protocol and the TCP protocol (bursty traffic). The application parameters are set as recommended by [77].

5.1.2.1 Video Transmission

A bidirectional video transmission is initiated by the sender (node n_S in all the scenarios). A video transmission is active until the end of the simulation period, starting at 130 sec. Therefore, the video flow is active for about 15 minutes in all the scenarios. Each frame of video data arrives at a regular interval, determined by the number of frames per second (fps). Each frame is decomposed into a fixed number of slices, and transmitted as single packets. The size of these packets/slices is distributed using the *Truncated Pareto* (Table 5-9) distribution.

Table 5-9 Properties of video transmission

Inter arrival time between the beginning of each frame (sec)	Deterministic (10 fps)
# of packets in a frame	Deterministic (1)
Size of a packet (bytes)	Truncated Pareto (location, L = 20 bytes, shape = 1.2, maximum size = 125 bytes)

When evaluating the performance, the following statistics are considered.

- *End-to-end delay (video delay) at the application layer*: The time taken to send a video application packet to a receiving node's application layer. This is measured from the time it is created to the time it is received. Therefore, it includes all the buffering and other transmission delays that occur at UDP, IP, MAC and PHY layers. This statistic shows the average end-to-end delay at both the source and the destination nodes. The *end-to-end delay* of the video flow is referred to the *video delay* in subsequent sections.
- *Coefficient of Variation (CV) of video delay*: This statistic is computed as the ratio of the standard deviation to the mean video delay.
- *Number of retransmission attempts per node*: Total number of retransmission attempts before the packet was successfully transmitted by the WLAN MAC. This statistic is computed by summing up all the retransmissions done by each

active node in the network and showing as the average retransmissions done per node.

- *Total number of data packets dropped at the WLAN MAC layer*[10]: The data packets dropped by the WLAN MAC. This can happen due to a failure of all retransmissions until the retry limit that is used by the WLAN MAC protocol.

5.1.2.2 Audio Conferencing Flow

The bidirectional audio conferencing is initiated by the sender (node n_S of all the scenarios) with characteristics as defined in Table 5-10. The G 723.1 codec is set with a codec rate of 5.3 kbps and a digital signal processing ratio of 1 with speech detection enabled.

Table 5-10 Properties of audio conferencing

Silence period of both calling and called party (sec)	Exponential (Mean = 0.65)
Talk spurt period of both calling and called party (sec)	Exponential (Mean = 0.352)
Audio codec used Frame Size (ms) Codec rate (kbps) DSP processing ratio Speech Activity Detection	G 723.1 30 5.3 1.0 Enabled
Audio packets per frame	Deterministic (1)
Compression delay (sec)	Deterministic (0.020)
Decompression delay (sec)	Deterministic (0.020)

When evaluating the performance of the audio flow, the jitter is considered in addition to the statistics used for the video flow.

- *Jitter of the audio flow:* If two consecutive packets leave the originating node with time stamps *t1* & *t2* and are received at the receiving node at time *t3* & *t4* respectively, then the jitter is computed as $|(t4 - t3) - (t2 - t1)|$. This statistic shows the average jitter at both the source and the destination nodes.
- *Audio End-to-end delay (audio delay) at the application layer:* audio delay is computed for an audio flow by summing up all the following delays.
  - The time taken to send an audio application packet to a receiving node's application layer.
  - Decompression delay defined when setting audio parameters.
  - Compression delay defined when setting the audio parameters.
  - Processing time to form the packet at the originator and the receiver. This is computed as "*audio frames per packet* x *DSP processing ratio* x *Frame Size (= 30 ms)"*.

Compared to the average *video delay*, the average *audio delay* consists of additional 3 delays of compression (i.e. 20 ms), decompression (i.e. 20 ms) and the processing (i.e. 30x2 = 60 ms) delays.

---

[10] In the OPNET simulator, this parameter consists of data packets dropped due to an overflow of higher layer buffers as well.

5.1.2.3 Single FTP Download

A single FTP download is initiated by the node n_S in all the scenarios. The size of the file is set to 1 MB. The node n_D acts as an FTP server while the node n_S is the FTP client. The application performance of the FTP download is compared using the following statistics.

- *FTP Download Response Time, DRT (sec):* This is measured from the time an FTP client application sends an FTP request to the server to the time it receives an FTP response packet. This includes the signaling delay for the connection setup and tear-down. This statistic is collected after the connection is closed and is measured at the FTP client.
- *Total number of TCP Timeout Counts:* Number of TCP retransmissions done after the expiry of retransmission timeouts. This statistic is collected at the FTP server.
- *Total number of TCP Fast Retransmission Counts:* Number of TCP retransmissions done after detecting a packet loss with the receipt of 3 DupACKs. This statistic is collected at the FTP server.
- *TCP Segment Delay (sec):* This is measured from the time a TCP segment is sent from the source TCP layer to the time it is received by the TCP layer in the destination node. This statistic is collected at the FTP client.

The TCP Reno version is used with the parameters listed in Table 5-11.

Table 5-11 TCP parameters

Maximum segment size (bytes)	2264
Receiver buffer size (bytes)	65535
Slow start initial count	1MSS
Delayed ACK mechanism	Segment based
Maximum ACK delay (sec)	0.002
DupACK Threshold	3
Fast Retransmit	Enabled
Fast Recovery	Reno
Initial RTO	3 sec
Maximum RTO	64 sec

5.1.2.4 HTTP Web Access

A web browsing user is configured to run on the node n_S. The node n_D is configured to run as the web server. After receiving the HTTP GET request, the server responds to the HTTP request with the additional references to embedded image files. The initial HTML page is referred to as the “main object” and each of the constituent objects referenced from the main object are referred to as an “embedded object”. In HTTP/1.1, persistent TCP connections are used to download the objects, which are located at the same server and the objects are transferred serially over a single TCP connection. Therefore, the TCP overhead of slow-start and congestion control occur only once per

connection. The distributions of the parameters for the web browsing traffic model are determined based on the survey of the literature on web browsing traffic characteristics [77] as listed in Table 5-12. When using HTTP web access, the same TCP parameters shown in Table 5-11 are used.

The application performance of the HTTP web access is compared using the following statistics, in addition to the WLAN and TCP statistics used in an FTP download.

- *Page Response Time (PRT):* Specifies the time required to retrieve the entire page with all the contained inline objects.
- *Object Response Time (ORT):* Specifies the response time for each in-lined object from the HTML page.

Table 5-12 Properties of HTTP web access

Main object size (bytes)	Truncated Lognormal (Mean = 10710 bytes, Std Dev = 25032 bytes, Maximum page size = 2 Mbytes)
Embedded object size (bytes)	Truncated Lognormal (Mean = 7758 bytes, Std Dev = 126168 bytes, Maximum page size = 2 Mbytes)
Number of embedded objects per page	Deterministic (5)
Page inter arrival time (sec)	Exponential (Mean = 30)

5.2 MF and SF Distribution Algorithms

This section details how IP packets are distributed at the IP layer when using the discovered RDM routing paths simultaneously. The aim is to assign each incoming flow to a path considering the remaining bandwidth of a path. This particular problem is similar to the well known allocation problem known as the *bin packing problem*.[78, 79] There are four known standard heuristics for the traditional *bin packing problem* to match the requests (items) to the resources available (capacity of the bin). They are,

- ***Best fit:*** This selects the bin with the smallest remaining capacity that can fit the item. The aim is to minimize the remaining capacity of the bin.
- ***Worst fit:*** This selects the bin with the largest remaining capacity that can fit the item. The aim is to maximize the remaining capacity of the bin.
- ***First fit:*** This selects the first bin that fits with the item capacity.
- ***Next fit:*** This packs the item in the currently opened bin if it has enough remaining capacity. If the item does not fit, the bin is closed and a new one is opened.

Table 5-13 compares the features of the standard *bin packing problem* and the flow distribution algorithm used to distribute IP packets of different flows.

According to previous work, the *best fit* and the *first fit* are the best algorithms for solving the *bin packing problem* [78] to optimize a number of bins to be used. However, two important differences should not be neglected when applying the *bin packing* algorithm to distribute IP packets of a flow. Firstly, the number of routing paths used for the distribution is limited to two since the RDM discovers only a maximum of 2 node disjoint routes. Secondly, a flow distribution algorithm should alleviate the congestion by maximizing the remaining bandwidth of a used RDM path. Due to these reasons,

despite its name, the *worst fit* algorithm appears to be the appropriate algorithm to be used to distribute flows.

Table 5-13: Comparison of bin packing problem vs flow distribution algorithm

Bin Packing Problem	**Flow Distribution Algorithm**
Objective is to arrange items in the least number of fixed sized bins by minimizing or maximizing the remaining capacity	Objective is to distribute flows among selected RDM paths by maximizing the remaining bandwidth of the paths
Size of items	Required bandwidth of the flows
Remaining capacity of bins	Remaining bandwidth of the RDM routing paths
Number of bins that can be used is not limited	Number of paths that can be used is limited
Sum of item sizes in a bin should not exceed the bin capacity	Available capacity of a path can be exceeded, but avoid this as far as possible

Table 5-14 Computation of required bandwidth of each flow

Application Flow	**ToS (FID)**	**Parameters used**	***ReqBW*, bytes/sec**
Video	Streaming multimedia (4)	$Mean_{frame\_size} \times number\ of\ frames$	$\frac{(1.2 \times 20)}{(1.2-1)} x10 = 1200$
Audio	Interactive voice (6)	$\frac{Codec\_rate \times Mean_{talk}}{Mean_{talk} + Mean_{silence}}$	$\left[\frac{(5300 \times 0.352)}{(0.65 + 0.352)}\right] \div 8 = 233$
FTP/HTTP	Best effort (0)	$MSS \times numberof\ segments$	$(2356bytes) \times 4 = 9424$

The standard *worst fit* algorithm allocates the next flow on the list to the path with the largest remaining bandwidth. If the incoming flows are usually of the same size, then this strategy might be very effective. Therefore, the *worst fit* algorithm has been modified by giving a higher priority to the flows having a higher required bandwidth. Then, the mapping of flows is done starting from the biggest flow to the smallest flow. When implementing the *worst fit* algorithm in the OPNET simulator, the following mechanisms are considered:

- *Dynamic measurement of the required bandwidth of a flow:* Since it is difficult to measure the required bandwidth of a flow dynamically, it is computed based on the application parameters assuming the computed bandwidth remains constant for the lifetime of the flow. A few examples for the computation of required bandwidth are given in Table 5-14.
- *Identifications of the packets of a single flow at the IP layer:* There are a number of ways to identify a flow uniquely as described in section 3.3.3. In the

implementation, *Type of Service* (ToS) of a flow is used to identify the packets of a flow at the IP layer.

- When implementing both the SF and the MF distribution algorithms, the following variables are used.
  - *Flow_List:* consists of the computed required bandwidth of a flow (*ReqBW*) and the *FlowID* (i.e. to keep ToS value of a flow).
  - Variables in *NextHopList* of RDM routing table (see Figure 3-5).
    - *OrginalBW (OrgBW):* keeps the PHY layer bandwidth
    - *RemainingBW (RemBW):* is computed by deducting the PL from the *OrgBW* (see Table 5-15).
    - *FID_List:* keeps the ToS value of already mapped flows.
    - *SentCount:* keeps the number of packets that are forwarded via a next-hop. This is used when SF distribution is enabled.

Table 5-15 Remaining bandwidth of each path as computed by RDM protocol

	Basic Topology		String Topology						Grid Topology	
			BTL-1		BTL-2		BTL-3			
	PL (%)	*RemBW* (%)	PL (%)	*RemBW* (%)	PL (%)	*RemBW* (%)	PL (%)	*RemBW* (%)	PL (%)	*RemBW* (%)
Primary Path	0	1	0.0328	0.9672	0.0928	0.9072	0.1856	0.8144	0	1
Secondary Path	0	1	0.0656	0.9344	0.1856	0.8144	0.3712	0.6288	0.0656	0.9344

5.2.1 MF Distribution Algorithm

In MF distribution, IP packets that belong to the same flow are forwarded via the same path. A flow is uniquely identified at the IP layer by the *ToS* value of a packet. The distribution of flows is done based on the *worst-fit* algorithm which is implemented to map the *ReqBW* of flows and the *RemBW* of paths. The path with the highest remaining bandwidth carries the flow with the largest required bandwidth. The MF distribution algorithm is implemented in the **ip_cmn_rte_table.c** external file.

Table 5-16 Example – Use of MF distribution algorithm in basic topology

	NextHopList [NexthopAddr, PID, RemBW, FIDList]	*FlowList* [Flow, FID, ReqBW]	Flow Allocation
1st iteration	node1, 1, 1 Mbps, - node7, 2, 1 Mbps, -	FTP, 0, 75392bps Video, 4, 9600bps Audio, 6, 1862bps	FTP is allocated to primary path (via node 1)
2nd iteration	node7, 2, 1 Mbps, - node1, 1, 924608bps, 0	Video, 4, 9600bps Audio, 6, 1862bps	Video is allocated to secondary path (via node 7)
3rd iteration	node7, 2, 990400bps, 4 node1, 1, 924608bps, 0	Audio, 6, 1862bps	Video is also allocated to secondary path (via node 7)
End	node7, 2, 988538bps, 4 &6 node1, 1, 924608bps, 0	-	FTP via node 1 Video & audio via node 7

As shown in Figure 5-6, the flow allocation is done for each incoming IP packet. When the lifetime of a flow expires (e.g., the completion of an FTP download) or a RDM path breaks, contents in the *FlowList* and the *NextHopList* are updated dynamically.

Therefore, the MF distribution algorithm always considers the currently active number of flows and paths.

Table 5-15 and Table 5-14 show the remaining bandwidth (*RemBW*) and required bandwidth (*ReqBW*) respectively. The *RemBW* is computed by the RDM protocol assuming that the original bandwidth is 1 Mbps.

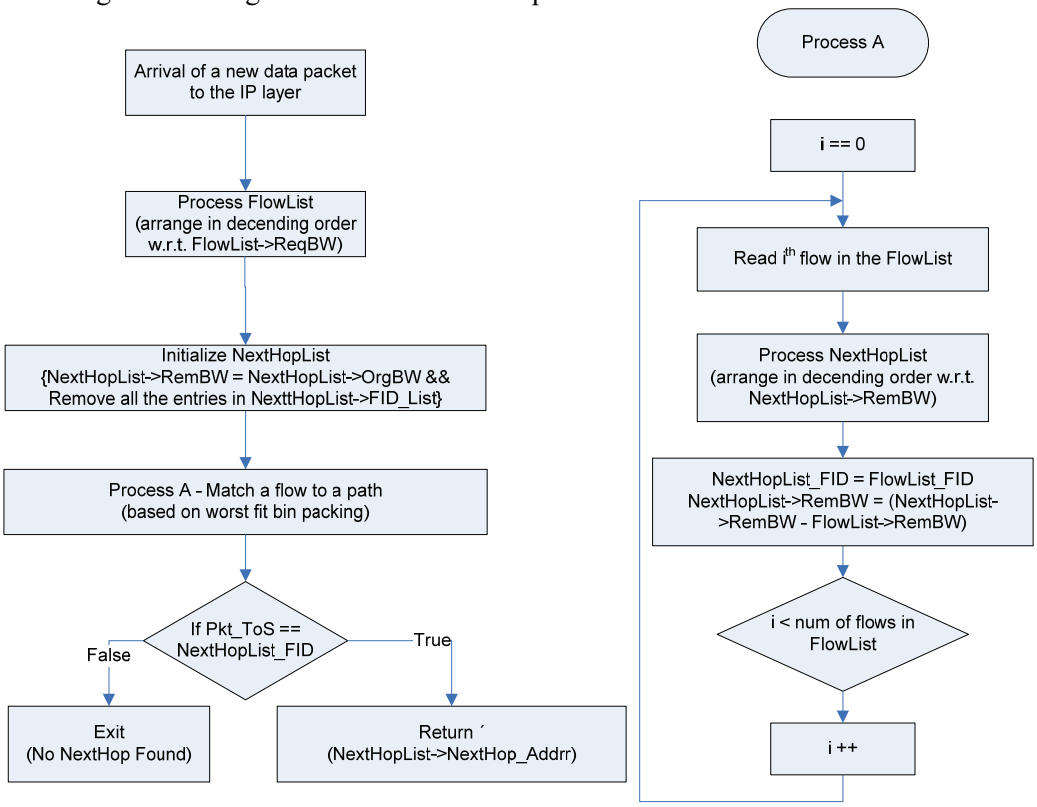


Figure 5-6 Flow chart – MF distribution algorithm

5.2.1.1 Example - MF Distribution

This section details how the MF distribution is done when using 3 flows: an audio conferencing session, an FTP download and a video transmission in the basic topology. The flows are distributed according to the MF distribution algorithm explained in section 5.2.1. The *ReqBW* is computed in bps as explained in Table 5-14. Table 5-16 shows how MF distribution is done in detail.

5.2.2 SF Distribution Algorithm

The SF distribution is applicable when only a single flow is active. This distribution splits more packets to the path with more *RemBW*. The code implemented in the **ip_cmn_rte_table.c** external file distributes IP packets in a round-robin fashion over 2

RDM paths. Each time when a new IP packet is forwarded to a given path, the *SentCount* value in the *NextHopList* is incremented by one. The next IP packet follows the same path until this value reaches a threshold named *PathWeight*, then the traffic is switched to the other path. The *PathWeight* of each path is computed considering the *RemBW*. The value of *SentCount* of both primary and secondary paths is set by default to zero. The *PathWeight* is computed as follows.

$$\text{if (PrimaryPath\_RemBW} \geq \text{SecondaryPath\_RemBW)}$$

$$\text{PrimaryPathWeight} = \frac{\text{PrimaryPath\_RemBW}}{\text{SecondaryPath\_RemBW}}$$

$$\text{SecondaryPathWeight} = 1$$

$$\text{if (SecondaryPath\_RemBW} \rangle \text{ PrimaryPath\_RemBW)}$$

$$\text{SecondaryPathWeight} = \frac{\text{SeconadryPath\_RemBW}}{\text{PrimaryPath\_RemBW}}$$

$$\text{PrimaryPathWeight} = 1$$

The *PrimaryPathWeight* and *SecondaryPathWeight* should be positive integers and the results of the above computation should therefore be rounded off to an integer.

5.3 Simulative Performance Analysis: MF Distribution

When transmitting both UDP and TCP data packets together (especially over SP), there are a higher number of packet losses that occur due to the hidden node problem. In given scenarios, it is very likely that TCP packets collide with other smaller audio and video packets since TCP packets are the largest packets and hence take more time to transmit. Section 10.4 of Appendix II discusses the effect of enabling and disabling RTS/CTS for different flows. It shows that transmission of larger packets (e.g. TCP-Data in the FTP download) together with smaller packets without enabling RTS/CTS deteriorates the TCP transmission drastically, especially for the SP. Therefore, all the results discussed in the rest of this chapter are taken setting the RTS threshold to 80 bytes. In summary, results of SP, FRDM and NRDM are compared by enabling RTS/CTS messages for all application packets.

All the results discussed in this section use the MF distribution when using the RDM paths simultaneously. All active flows are distributed via one path when using the SP. Before taking the results of different scenarios, an applicability of the proposed MF distribution algorithm is investigated in detail. For this purpose, a simple network of SP and FRDM scenarios in the basic topology are used with audio, video and an FTP download. By changing the *ReqBW* of a flow, the following 4 distributions are analyzed in detail.

- *Case 1:* This uses the proposed MF distribution algorithm. It forwards FTP via the primary path and the other flows (audio and video) via the secondary path. This distribution is done as explained in Table 5-16.
- *Case 2:* Video via the primary path, FTP and audio via secondary path. This distribution is done by assigning a higher *ReqBW* for video and a lower *ReqBW* for FTP.

- *Case 3:* Audio via the primary path, FTP and video via the secondary path. This distribution is done by assigning a higher *ReqBW* for audio and a lower *ReqBW* for FTP.
- *Case 4:* This uses the proposed MF distribution algorithm as in Case 1. But, after the completion of the FTP download, flows are re-allocated. After re-allocation, audio and video flows are separated by changing the video flow to the primary path since the FTP download has been completed.

The performance of the above cases is compared with the SP scenario (Figure 5-1 (a)). When using SP, all flows are forwarded via one path.

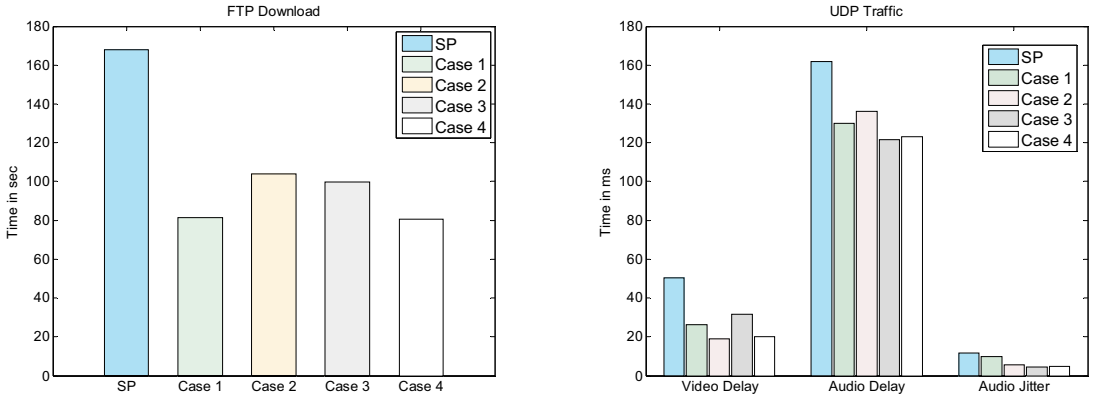


Figure 5-7 Application performance with different MF distribution algorithms : SP, case 1(FTP vs "video & audio"), case 2 (audio vs "FTP & video"), case 3 (audio vs "FTP & video") and case 4 (FTP vs "video & audio" with dynamic flow distribution)

Analysis of results in Table 5-17 together with Figure 5-7 concludes the following.

- FTP performs better than in SP (sending all flows on one path) when using the RDM paths with MF distribution. The FTP download is faster when sending FTP alone (case 1 and case 4). FTP performance degrades when sending with UDP based traffic due to the capture condition that favors more frequently transmitted audio and video packets.
- The video flow shows the best performance when sending video alone (case 2) and similarly, the best audio performance can be achieved by sending audio alone (case 3).
- Comparing all different distribution methods, it shows that the performance of all applications improve compared to SP. When separating the FTP flow at the beginning and re-allocating the audio and video flows after the completion of the FTP download, the performance of all applications improve significantly as shown in case 4 of Table 5-17.

Table 5-17 Application performance with different MF distribution algorithms

Scenario	FTP	Video Transmission		Audio Conferencing		
	DRT, sec	delay, ms	CV	Jitter, ms	delay, ms	CV
SP	168.10	50.20 ±96.96	1.931	11.76 ±24.09	161.62 ±120.46	0.745
Case 1 FTP vs "video & audio"	81.26	26.28 ±22.86	0.869	9.71 ±12.14	130.10 ±26.83	0.206
Case 2 video vs "FTP & audio"	103.91	18.93 ±15.06	0.795	5.59 ±18.42	136.13 ±75.58	0.555
Case 3 audio vs "FTP & video"	99.65	31.78 ±64.13	2.017	4.56 ±9.56	121.69 ±18.14	0.149
Case 4 FTP vs "video & audio"	80.62	20.21 ±18.91	0.935	4.94 ±10.14	123.06 ±22.26	0.180
Performance gain of case 4, compared to SP	52%	59.74%	51.57%	57.99%	23.85%	75.83%

According to the above results, the MF distribution improves the application performance for all cases. These results show that the transmission of TCP based traffic together with UDP based traffic with higher sending rate affects the TCP traffic badly mainly due to the capturing of wireless media by the UDP traffic. Therefore, rest of the scenarios is run by not sending the TCP based flows together with audio and video flows. Once the FTP download or the HTTP web access is over, re-allocation of the remaining active flows is done and the rest of audio and video flows are separated among the RDM paths. However, the MF distribution algorithm proposed in this work can be used to separate any flow or a set of flows to enhance the performance by assigning higher values to the *ReqBW*.

The subsequent sections analyze the use of the proposed MF distribution algorithm when using RDM routes in different topologies. The performance gain of each parameter is computed w.r.t. the performance of the SP. The performance of the SP is analyzed by sending all the active flows over the path with the lowest hop count.

5.3.1 Basic Topologies

The SP, FRDM and NRDM scenarios in the basic topology are used to simulate the use of four flows. At the beginning, a video transmission, an audio conference and an FTP download are initiated. After the completion of the FTP download, a HTTP web browsing session is initiated. The HTTP, video and audio flows are active until end of the simulation period.

Table 5-18 shows the statistics collected when initiating the above mentioned four flows by the originating node, n_S. When using FRDM and NRDM routes in the basic topology, active flows are distributed among the upper and lower paths (Figure 5-1). In this scenario, when using the FRDM and NRDM routes, both FTP and HTTP flows are directed over the primary path and the other two flows are directed over the secondary path. All application flows enjoy the benefits of SUM routing when distributing flows among FRDM routes. The performance of both the video and audio flows shows better performance (Figure 5-8) compared to the use of all flows on the SP, even after separating the TCP based traffic on the other interfering path (NRDM scenario).

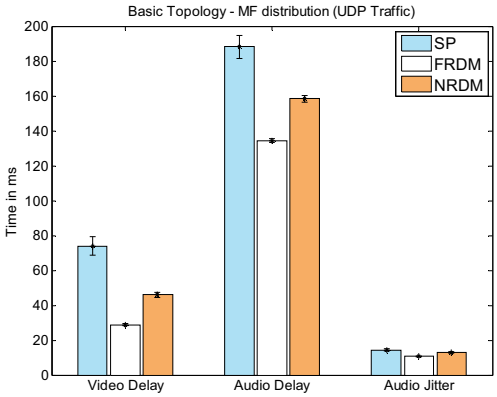


Figure 5-8 Basic topologies - MF distribution ("audio & video" vs "FTP & HTTP")

Table 5-18 Basic topologies - MF distribution ("audio & video" vs "FTP & HTTP")

Scenario	Video Transmission		Audio Conferencing			FTP download	HTTP web access	
	Delay, ms	CV of Delay	Jitter, ms	Delay, ms	CV of Delay	DRT, sec	PRT, sec	ORT, sec
SP	73.89 ±5.269	1.789 ±0.0651	14.41 ±0.636	188.22 ±6.67	0.799 ±0.0208	154.71 ±6.49	6.84 ±0.881	2.35 ±0.302
FRDM	28.88 ±0.745	1.061 ±0.0483	10.70 ±0.285	134.41 ±1.02	0.2830 ±0.0171	81.98 ±3.69	4.75 ±0.524	1.86 ±0.340
Perf. Gain	60.91%		25.74%	28.58%		47.01%	30.55%	20.85%
NRDM	46.07 ±1.583	1.646 ±0.0338	12.80 ±0.408	158.33 ±1.88	0.600 ±0.0172	158.95 ±7.66	7.69 ±1.708	3.04 ±0.505
Perf. Gain	37.65%		11.17%	15.88%		-2.74%	-12.43%	-29.36%

Table 5-19 Basic topologies - MF distribution (data packets dropped by WLAN)

Scenario	Data Packets dropped by WLAN
SP	887 ±115
FRDM	200 ±17
NRDM	926 ±50

However, both FTP and HTTP application performance (Figure 5-9 and Table 5-18) degrade even though these two applications are not sent together with UDP based flows in the NRDM scenario. In general, the NRDM scenario shows the highest amount of dropped packets (Table 5-19) due to the use of interfering routes simultaneously. The simultaneous use of interfering routes causes more congestion in the nodes carrying TCP traffic due to the capture condition created by the nodes on the other path, that are transmitting UDP traffic more frequently (see section 10.2.2.1 of Appendix II).

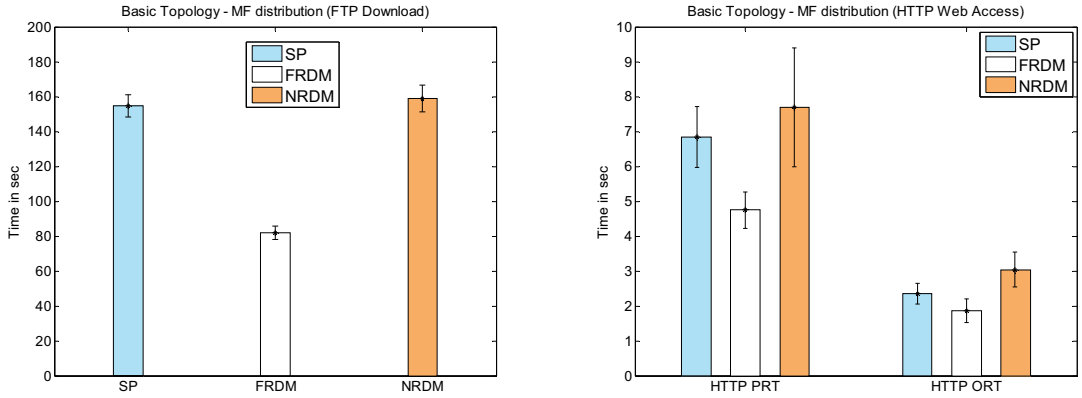


Figure 5-9 Basic topologies - MF distribution (FTP and HTTP)

5.3.2 String Topology

The string topology (Figure 5-2) is set up with three paths, each with different hop counts and different effect of the BTL. The SP itself carries the BTL. The upper path consists of 6 intermediate nodes, the middle path consists of 5 intermediate nodes and the lower path is configured with 7 intermediate nodes. The middle path which is used as the SP carries a BTL of 6.56% (denoted as BTL-1 in Table 5-4). The FRDM paths selected are the lower (i.e. primary) and the upper (secondary) paths.

Three flows, namely the FTP download, the audio conferencing and the video transmission are considered in this scenario. Table 5-20 together with Figure 5-10 shows the statistics collected when distributing these flows in the string topology.

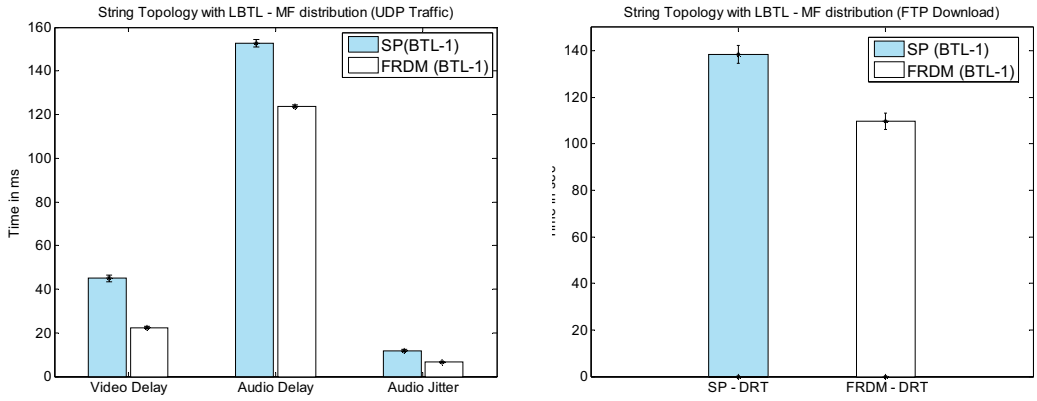


Figure 5-10 String topology with LBTL- MF distribution (“audio & video” vs “FTP”)

Table 5-20 String topology LBTL - MF distribution (“audio & video” vs “FTP”)

Scenario	Video Transmission		Audio Conferencing			FTP download	WLAN Packets dropped
	Delay, ms	CV	Jitter, ms	Delay, ms	CV	DRT, sec	
SP (BTL-1)	44.94 ±1.453	2.118 ±0.044	11.90 ±0.413	152.62 ±1.711	0.705 ±0.0215	138.31 ±3.92	579 ±20
FRDM (BTL-1)	22.42 ±0.504	0.995 ±0.046	6.50 ±0.157	123.80 ±0.465	0.234 ±0.0119	109.68 ±3.46	318 ±34
Perf. Gain	50.11%		45.33%	18.88%		20.69%	45.07%

Table 5-20 shows that there is a significant performance improvement in each application flow when distributing among FRDM routes using the proposed distribution algorithms. In this distribution, the FTP uses the primary path alone, while the video and audio flows are distributed over the secondary path. After the completion of the FTP download, the dynamic nature of the MF distribution algorithm permits to reevaluate and redistribute packets that are still in transit (audio and video). After the reevaluation, video packets are sent on the primary path and audio packets on the secondary path until the end of simulation duration.

5.3.3 Grid Topology

As explained in section 5.1.1.3, the grid topology selects P_1 as the SP. P_1 and P_3 are used as FRDM routes and P_1 and P_2 are NRDM routes. In contrast to the string topology, the SP (P_1), which is also selected as the primary path of the RDM paths, is not carrying any BTL. The secondary path (P_3) is influenced by the unidirectional BTL generated from node, n_6 to node, n_7 and from node, n_8 to node, n_9.

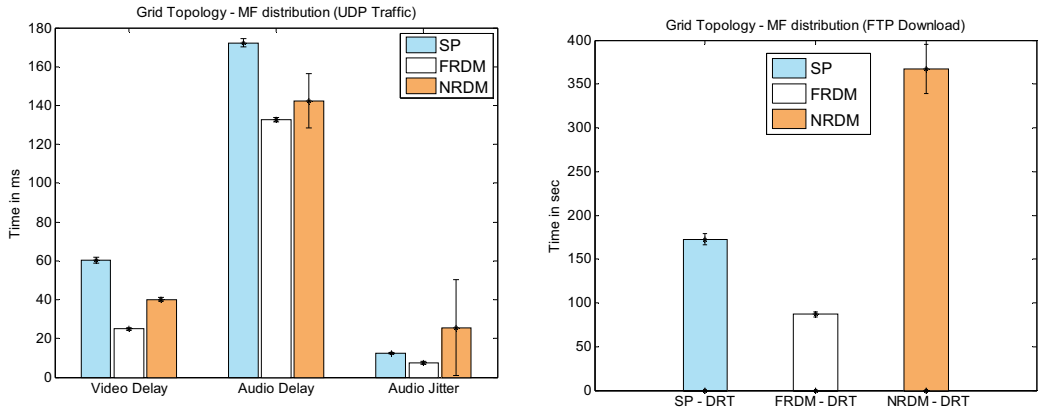


Figure 5-11 Grid topology – MF distribution (“audio & video” vs “FTP”)

When using FRDM and NRDM routes, the primary path is reused for the distribution of video packets, while audio packets still use the secondary path after the completion of

the FTP download. The outcome of the grid topology is also similar to the basic topology. Compared to the basic topology, these results (Table 5-21 together with Figure 5-11) show that the use of FRDM routes to distribute multiple flows improves the application performance significantly, even though the secondary path is influenced by the BTL.

When using NRDM routes, FTP DRT increases significantly due to the effect of the other flows (UDP based traffic) on the other interfering path. This occurs mainly due to the capture of media by the UDP based traffic on the secondary path. This affects the TCP traffic that goes over the primary path since the secondary path is interfering with the primary path. Therefore, most of the TCP-Data segments are dropped due to unsuccessful RTS/CTS handshake (see Figure 10-4 of Appendix II). Furthermore, though both the video and audio flows show better performance with the NRDM routing, most of the packets are dropped when using NRDM routes as in the basic topology. The video and audio packets can travel faster due to the alleviation of the congestion (compared with the SP routing) in both FRDM and NRDM scenarios.

Table 5-21 Grid topology - MF distribution ("FTP" vs "audio & video")

Scenario	Video Transmission		Audio Conferencing			FTP download	Data Packets Dropped
	Delay, ms	CV	Jitter, ms	Delay, ms	CV	DRT, sec	
SP	60.36 ±1.457	1.822 ±0.0482	12.41 ±0.270	172.25 ±2.155	0.732 ±0.0266	172.50 ±6.236	622 ±59
FRDM	25.04 ±0.589	1.157 ±0.0658	7.35 ±0.757	132.69 ±1.305	0.281 ±0.0229	86.90 ±2.76	369 ±80
Perf. Gain	58.51%		40.77%	22.96%		49.62%	40.67%
NRDM	40.12 ±0.977	1.297 ±0.1956	25.62 ±24.768	142.46 ±13.861	0.511 ±0.0949	367.03 ±28.05	897 ±10
Perf. Gain	33.53%		-106%	17.29%		-112.77%	-44.21%

5.3.4 Random Topology

The random topology is used to distribute 4 flows, namely, audio, video, HTTP web access and FTP download as in the basic topology. When using the RDM paths, the video and audio flows are directed over the secondary path and the primary path is used to send all TCP traffic.

Table 5-22 together with Figure 5-12 shows the statistic collected in the random topology with MF distribution. As in the basic topology, distribution of multiple flows among RDM paths shows significant improvement in all used applications due to alleviating the congestion on one path. Compared to the results shown in Table 5-18, the delays computed in the random topology is lower for both the SP and the RDM scenarios, since the random topology discovers the paths with less number of hops.

Table 5-22 Random Topology - MF Distribution (“audio & video” vs “FTP & HTTP”)

Scenario	Video Transmission		Audio Conferencing			FTP download	HTTP web access	
	Delay, ms	CV of Delay	Jitter, ms	Delay, ms	CV of Delay	DRT, sec	PRT, sec	ORT, sec
SP	33.389 ±7.852	2.419 ±0.219	8.421 ±2.865	135.020 ±9.792	0.586 ±0.038	109.314 ±26.224	4.936 ±0.758	2.289 ±0.288
RDM	22.995 ±1.608	2.072 ±0.237	7.296 ±0.447	125.319 ±2.218	0.416 ±0.053	84.481 ±20.085	3.754 ±0.909	1.917 ±0.416
Perf. Gain	31.13%		13.35%	7.18%		22.71%	23.94%	16.25%

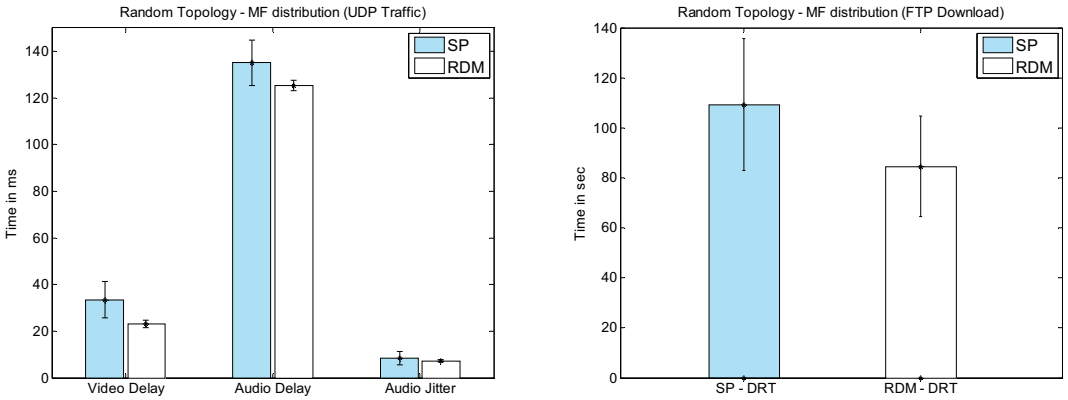


Figure 5-12 Random topology – MF distribution (“audio & video” vs “FTP & HTTP”)

5.3.5 Mobility Scenario

A mobility scenario is configured as shown in Figure 5-5. Both video and audio flows are evaluated by distributing two flows among RDM routes.

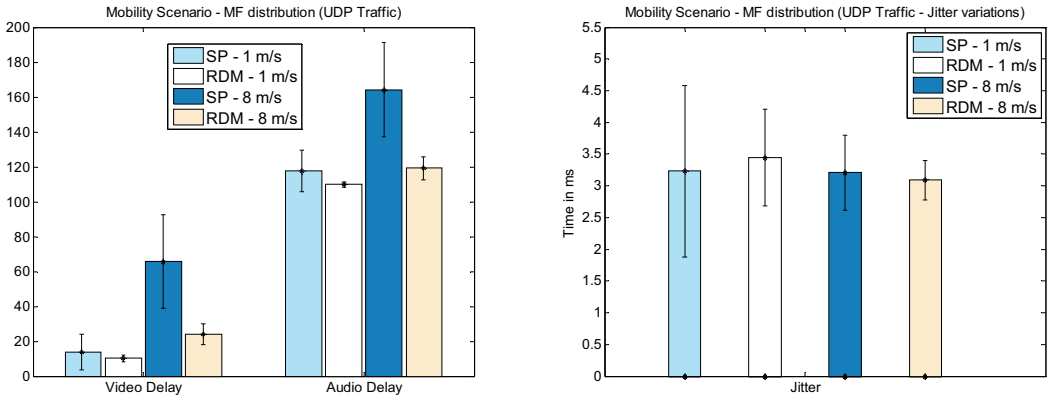


Figure 5-13 Mobility Scenario – MF Distribution (“audio vs video”)

This scenario is run for 2 different speeds of mobility, one to simulate the walking speed (1 m/s) and the other to simulate a higher speed of mobility with 8 m/s.

Table 5-23 together with Figure 5-13 shows the evaluated application parameters. The application performance degrades with the increase of mobility for both SP and RDM scenarios. The number of packets dropped at the WLAN layer is significantly increased with the increase of mobility. There is a performance improvement in all evaluated parameters when distributing the audio and video flows among RDM routes irrespective of the type of mobility. This happens due to alleviating the congestion on one path and also due to less route discoveries.

Table 5-23 Mobility scenario - MF distribution ("audio vs video")

Scenario	Video Transmission		Audio Conferencing			WLAN Layer Packets dropped
	Delay, ms	CV	Jitter, ms	Delay, ms	CV	
SP –1 m/s	13.81 ±10.13	7.60 ±4.16	3.23 ±1.357	117.62 ±11.95	1.34 ±1.21	626 ±265
RDM –1 m/s	10.33 ±1.98	5.03 ±1.62	3.45 ±0.76	109.79 ±1.68	1.65 ±1.47	466 ±120
SP –8 m/s	65.89 ±26.87	10.69 ±2.58	3.21 ±0.590	164.14 ±26.99	3.83 ±1.84	4520 ±832
RDM –8 m/s	24.11 ±5.98	10.23 ±3.49	3.09 ±0.31	119.17 ±6.61	1.58 ±0.79	4477 ±380

Table 5-24 Mobility scenario - MF distribution (route discovery parameters)

Scenario	RREQs sent	Successful Route Discoveries		RERRs sent	Average RDT, ms		
SP –1 m/s	10 ±6	10 ±6		58 ±41	1094.71 ±958.01		
SP –8 m/s	46 ±6	42 ±5		399 ±97	1511.35 ±524.36		
						% of usage of routes	
		SP	RDM			SP	RDM
RDM –1 m/s	4 ±1	1 ±0	3 ±1	49 ±16	399.02 ±125.50	59 ±11	41 ±11
RDM –8 m/s	23 ±4	4 ±1	17 ±2	311 ±63	443.59 ±177.46	65 ±10	35 ±10

Table 5-24 shows the route discovery parameters detailed in section 5.3.5. The RDM protocol uses 2 routes and it does not initiate a route discovery until both of the routing paths expire. Therefore, the RREQs and RERRs generated when using the RDM routes are less compared to the SP. The following conclusions can be made with the increase of mobility.

- Both the SP and RDM protocols have to disseminate more RREQs to discover routes with the increase of mobility. Some RREQs do not reach the destination with the increase of mobility and therefore the successful route discoveries are lower than compared to the number of RREQs generated.

- The RDM protocol tends to discover more SP routes as shown in successful route discoveries in Table 5-24, since the destination receives less RREQs to evaluate to find disjoint paths with the increase of mobility.
- It also shows that the percentage of simultaneous usage of RDM routes decreases with the increase of mobility from 1 m/s to 8 m/s (41% to 35%).

5.3.6 Performance Analysis: SF distribution when multiple flows are present

The previous section details how flows are distributed individually when more than one flow is present. This section focuses on evaluating the performance when splitting packets of an individual flow while multiple flows are generated. This section shows only a few selected scenarios (basic topology and string topology) from the previous section and compares them with the MF distribution results shown earlier. Only FRDM scenarios are chosen and the results are compared in Table 5-25. FRDM-MF shows the results from the previous section while distributing individual flows according to the proposed MF distribution algorithm. FRDM-Split shows the results while splitting packets of individual flows together with other existing flows.

When splitting packets of an individual flow together with TCP based flows, UDP based communication does not perform well compared to FRDM-MF results (i.e. separating UDP and TCP based flows under MF distribution algorithm). It shows higher delays and higher variations in the measured audio and video parameters. Splitting both TCP and UDP based packets simultaneously causes a higher number of packets to be dropped at the WLAN MAC layer (Table 5-26). As in the basic topology, the string topology (Table 5-27) also shows the same behavior for both audio and video flows, when splitting packets of an individual flow together with TCP based flows.

Table 5-25 FRDM scenario in basic topologies – MF vs SF distribution when multiple flows are present (UDP and TCP parameters)

Scenario	Video Transmission		Audio Conferencing			FTP download	HTTP web access	
	Delay, ms	CV of Delay	Jitter, ms	Delay, ms	CV of Delay	DRT, sec	PRT, sec	ORT, sec
FRDM – MF	28.88 ±0.745	1.061 ±0.0483	10.70 ±0.285	134.41 ±1.02	0.2830 ±0.0171	81.98 ±3.69	4.75 ±0.524	1.86 ±0.340
FRDM – Split	31.22 ±2.03	1.796 0.1271	9.97 ±0.796	133.82 2.33	0.457 0.0342	68.29 4.28	3.89 0.946	1.41 0.451

Table 5-26 FRDM scenario in basic topologies – MF vs SF distribution when multiple flows are present (Data packets dropped)

Scenario	Data Packets dropped @ WLAN
FRDM - MF	200 ±17
FRDM - Split	376 ±47

In contrast to UDP performance, TCP based traffic shows an improved performance when splitting TCP flows together with UDP based flows. This is analyzed further in the basic topology by investigating how packets are dropped on a selected set of nodes.

Table 5-27 String topology with LBTL – MF vs SF distribution when multiple flows are present (UDP and TCP parameters)

Scenario	Video Transmission		Audio Conferencing			FTP download	WLAN Packets dropped
	Delay, ms	CV	Jitter, ms	Delay, ms	CV	DRT, sec	
FRDM (BTL-1) - MF	22.42 ±0.504	0.995 ±0.046	6.50 ±0.157	123.80 ±0.465	0.234 ±0.0119	109.68 ±3.46	318 ±34
FRDM (BTL-1)- Split	26.32 ±1.121	1.696 ±0.174	9.33 ±0.422	127.69 ±1.376	0.518 ±0.3984	77.75 ±4.79	463 ±25

Table 5-28 shows how packets are dropped by the upper and lower paths:

- Upper path via node 1 and node 6 (see Figure 5-1)
- Lower path via node 7 and node 12 (see Figure 5-1)

When using MF distribution, the lower path is selected to send TCP flows while the upper path is selected to send UDP flows. When using SF distribution, packets of an individual flow are split to both paths identically. Therefore, Table 5-28 shows that a higher number of retransmission attempts and dropped data packets occur in the upper path (from node 1 and node 6) when using MF distribution due to a higher contention that is experienced with UDP flows. When splitting packets of an individual flow, dropped packets are distributed equally among two paths. Though there is a lower number of total dropped packets with the MF distribution, it shows that all the TCP-data segments are dropped by node D and node 12.

Table 5-28 FRDM scenario in basic topologies - Number of retransmissions and data packets dropped by the selected nodes

Nodes		Node S	Node 7	Node 1	Node 6	Node 12	Node D
No: of retransmission attempts	MF	7277	850	6598	6329	787	6293
	Splitting	5007	3509	3527	3470	4355	4463
No: of data packets lost	MF	2	2	12	42	8	17
	Splitting	16	21	38	26	27	35
No: of TCP-Data segments lost	MF	0	0	0	0	7	15
	Splitting	0	3	10	2	0	4

When sending the whole TCP flow on the lower path (via node 12), TCP-Data segments have to wait for a longer time to access node 12 on the lower path due to frequent UDP data transmissions on the other path via node 6. TCP-Data segments sent by node D can access next hops (both node 6 and node 12) more frequently when splitting TCP with UDP data. Splitting makes it possible for end nodes to access the nodes in both paths equally.

5.3.7 Performance Analysis: Standard SP vs SP discovered by RDM routing

This section compares the performance of applications when sending over the standard SP route and over the SP discovered by the RDM protocol. The RDM protocol uses the primary path as the SP while the standard DYMO uses the path with the lowest hop count as the SP. In case of the basic topologies (Figure 5-1) and the grid topology (Figure 5-3) used in the simulations, both the standard SP and the SP discovered by RDM are similar. Therefore, the string topology (Figure 5-2) is used to compare the performance of applications when using the following SP routes:

- *Standard SP:* The middle path with the lowest hop count is selected. Though the middle path has the lowest number of hops (5 hops) in this topology, it carries BTL as shown in Figure 5-2.
- *SP-RDM:* The lower path with the least interference and the least congestion is selected. Though this path has the highest number of hops (7 hops) in this topology, it does not carry any BTL.

Table 5-29 String topology with varying BTL - MF distribution ("audio & video") over SP vs SP-RDM

Scenario	Video Transmission		Audio Conferencing			WLAN Packets dropped
	Delay, ms	CV	Jitter, ms	Delay, ms	CV	
SP (BTL-1)	20.92 ±1.35	0.763 ±0.045	9.06 ±0.76	121.40 ±1.08	0.143 ±0.011	79 ±11
SP-RDM (BTL-1)	27.55 ±1.24	0.775 ±0.033	11.20 ±0.63	134.11 ±1.15	0.279 ±0.235	19 ±9
SP (BTL-2)	32.73 ±3.10	1.211 ±0.148	17.25 ±0.99	148.08 ±2.87	0.863 ±0.016	262 ±23
SP-RDM (BTL-2)	29.97 ±1.96	0.817 ±0.083	12.56 ±0.63	137.99 ±1.23	0.313 ±0.232	167 ±10
Perf. Gain	8.4%	-	27.19%	6.78%	-	36.25%

Both the audio and video flows are sent over the above selected SP and the simulation is repeated by changing the BTL. The BTL is changed from BTL-1 (6.56%) to BTL-2 (18.56%).

Table 5-29 together with Figure 5-14 shows how the parameters of applications vary when choosing different SP routes. The standard SP with a lower BTL of 6.56% (BTL-1) performs better than SP chosen by the RDM protocol (SP-RDM). SP-RDM shows higher delays due to the use of the extra 2 hops to send data over SP-RDM compared with sending data over the standard SP route. In this situation, the BTL that exists on the SP route does not degrade the other data communication. With the increase of BTL from 6.56% to 18.56%, the results show that SP-RDM performs better than the standard SP in terms of lower jitter (reduced by 27.19%) and a lower number of data packets dropped (reduced by 36.25%).

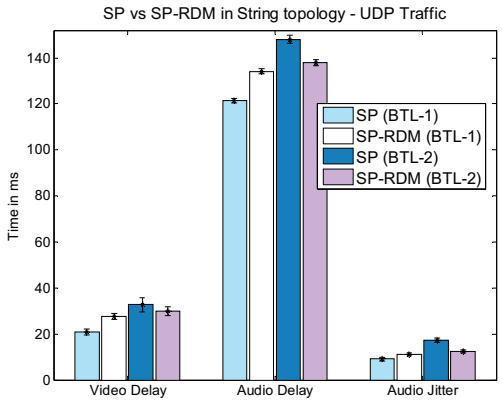


Figure 5-14: String topology with varying BTL - MF distribution ("audio & video") over SP vs SP-RDM

The data sent over the shortest path has to be sent via the congested nodes due to the BTL. Therefore, the use of the least congested path (SP-RDM (BTL-2)) gives better performance, though it has more hops than the congested path with the lowest hop count (SP (BTL-2)). A higher variation in both audio and video delays is shown when using the standard SP with the increase of the BTL.

5.4 Simulative Performance Analysis: SF Distribution

This section details the distribution of an individual flow among the RDM routing paths. When using the RDM routes, the packets of an individual flow is distributed as explained in section 5.2.2, distributing more packets on to the path which has a lower PL. All the packets of a flow are forwarded via the same path when using the SP route.

5.4.1 Basic Topology

Since two paths are identical in the basic topology (Figure 5-1), the distribution ratio is computed as 1:1 when using the FRDM and the NRDM routes. All packets of a single flow are forwarded via the upper path, when using the SP.

Table 5-30 Basic topologies - SF distribution ("audio conferencing")

Scenario	Jitter, ms	Audio Conferencing Delay		WLAN Layer	
		Average, ms	CV	Data Packets dropped	Retransmission attempts
SP	2.20 ±0.051	117.84 ±0.079	0.0375 ±0.000505	0	1549 ±68
FRDM	1.35 ±0.018	116.20 ±0.264	0.0246 ±0.000205	0	440 ±7
Perf. Gain	38.63%	1.39%		-	71.59%
NRDM	2.14 ±0.028	117.28 ±0.0336	0.0331 ±0.00045	0	816 ±17
Perf. Gain	2.72%	0.47%			47.32%

Table 5-30 together with Figure 5-15 shows the evaluated parameters when using an audio flow. The FRDM shows a 38% improvement in the jitter while the audio delay has been improved only by 1.39%. The SP carries twice the number of packets than each individual path in the FRDM scenario. Figure 5-16 compares the total number of packets transmitted by node 1 in the upper path of the basic topologies when using SP and FRDM routes. Since there are more packets on the SP, more retransmission attempts are done in the SP than in the FRDM scenario. The CV of the audio delay shows the variation of individual delays is lower when splitting the audio packets among the FRDM paths. Though the NRDM does not show any significant performance gain like in the FRDM routes, the NRDM also performs better than the SP since it helps to alleviate the congestion in one path.

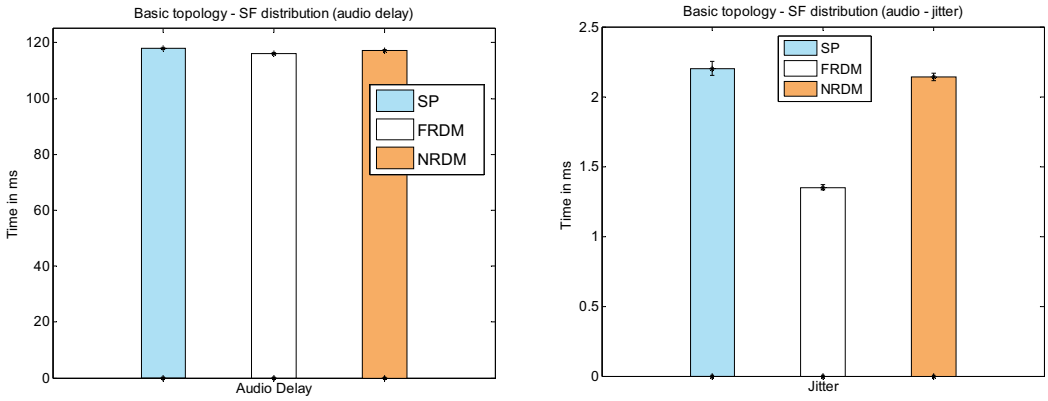


Figure 5-15: Basic topologies - SF distribution (“audio conferencing”)

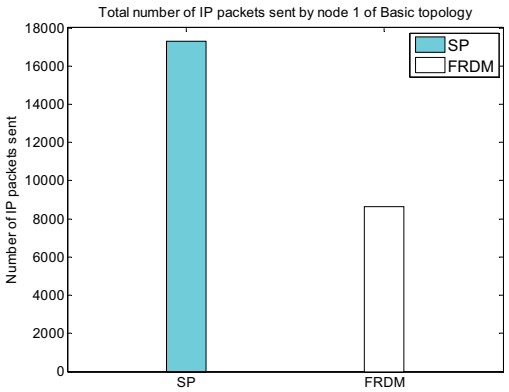


Figure 5-16: Traffic sent by node 1 of basic topologies when using the SP and the FRDM routes

Table 5-31 together with Figure 5-17 presents the statistics collected with a video transmission flow. Usually all the scenarios are run by setting the RTS threshold to 80 bytes. This scenario is repeated by changing the RTS threshold to 140 bytes as well.

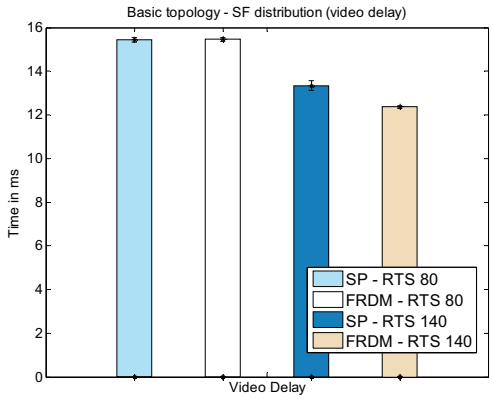


Figure 5-17: Basic topologies – SF distribution (video delay)

Table 5-31 Basic topologies - SF distribution ("video transmission")

Scenario	Video Delay		WLAN layer	
	Average, ms	CV	Data Packets dropped	Retransmission attempts
SP - RTS 80	15.44 ±0.105	0.21811 ±0.00339	0	4 ±2
FRDM - RTS 80	15.46 ±0.086	0.02253 ±0.00560	0	42 ±17
SP -RTS 140	13.34 ±0.232	0.32926 ±0.02468	0	499 ±145
FRDM - RTS 140	12.36 ±0.054	0.306357 ±0.005657	0	33 ±18
Perf. Gain	7.34%	6.95%	-	93.38%

The performance of FRDM does not show any gain while enabling RTS with 80 bytes. By increasing the RTS threshold to 140 bytes, the FRDM shows better performance. As explained in section 10.4, RTS with 80 bytes enables all video packets to be transmitted with the RTS/CTS handshake. The video packet sizes are *truncated Pareto* distributed between 101 bytes to the maximum size of 205 bytes. All video packets are transmitted with RTS/CTS handshake when RTS threshold is set to 80 bytes. Therefore, the packets have to wait longer before sending out at each node. Here, the delays for the FRDM and the SP are approximately equal. In FRDM, video packets reach the end nodes (n_1, n_7, n_6 & n_{12}) more or less at the same time via the lower and upper paths. These nodes try to access the source or the destination nodes simultaneously. This competition between end nodes of different paths causes an additional delay. Further analysis of retransmission attempts shows that 99% of the retransmission attempts are done by the end nodes, n_1, n_7, n_6 & n_{12}. When setting the RTS threshold to 140 bytes, most of the video packets are transmitted without the RTS/CTS handshake. This makes the video packets to transmit sooner (i.e. with a lower delay) showing the lower delay compared to the delay in the SP. The retransmission attempts in the SP are significantly increased with the increase of the RTS threshold due to the collisions of packets at the WLAN layer, when transmitting data with RTS/CTS handshake.

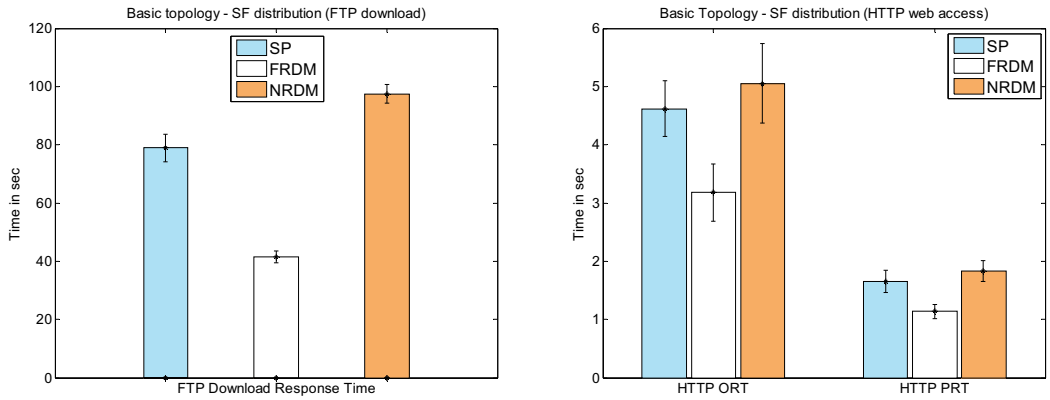


Figure 5-18: Basic topologies – SF distribution ("FTP download and HTTP web access")

Comparing with the performance gain of audio, improvement by splitting the video flow among identical paths is lower since video flow in the SP itself does not create much congestion at the WLAN layer. The media access delay is increased if packets are sent at a higher rate though the packet sizes are smaller. For example, the audio packets are sent at about 30 packets per second (exponentially distributed) while the video packets are sent at a constant rate of 10 packets per second.

Table 5-32 Basic topologies - SF distribution ("FTP download")

Scenario	FTP Download		Retransmissions @ TCP	
	DRT, sec	Segment delay, ms	Due to timeouts	Due to DupAcks
SP	78.95 ±4.71	238.52 ±5.35	14 ±2	19 ±2
FRDM	41.57 ±2.05	235.05 ±7.09	4 ±1	16 ±2
Perf. Gain	47.34%	1.45%	71.42%	
NRDM	97.42 ±3.20	195.45 ±2.36	26 ±1	17 ±2
Perf. Gain	-23.39	18.05	-85.71	10.52

Table 5-32 and Table 5-33 together with Figure 5-18 show a significant performance improvement when using the FRDM routes to distribute both FTP and HTTP flows. The DRT, the PRT, the ORT and the total number of retransmission attempts are considerably reduced. There is also a slight reduction of the segment delay due to the alleviation of congestion by distributing the packets.

Though the NRDM shows better performance for splitting UDP traffic (Table 5-30 & Table 5-31), the performance of both FTP and HTTP degrade when splitting traffic among interfering paths (NRDM routes). The capture of media by the nodes in one path causes more TCP packets (either TCP-Data or TCP-Ack) to be dropped continuously.

How the capture of media affects the TCP throughput is explained in Figure 10-6 of Appendix II.

Table 5-33 Basic topologies - SF distribution ("HTTP web access")

Scenario	PRT, sec	ORT, sec	Segment delay, ms	Num. of TCP retransmission counts
SP	4.62 ±0.477	1.65 ±0.194	196.43 ±11.83	41 ±7
FRDM	3.18 ±0.492	1.14 ±0.122	190.48 ±8.21	24 ±6
Perf. Gain	31.16%	30.91%	3.03%	41.46%
NRDM	5.054 ±0.678	1.83 ±0.179	160.22 ±9.28	39 ±14
Perf. Gain	-9.39%	-10.90%	18.43%	4.87%

5.4.2 String Topology

As explained in section 5.1.1.2, the use of FRDM routes in the string topology should distribute packets of a single flow based on 2:1 ratio. That means that it should send more packets on the lower path (primary path) than the upper path (secondary path). When using the SP, all packets of the flow are forwarded via the middle path, which is also congested with the BTL (Figure 5-2).

This topology is evaluated by varying the BTL between the nodes n_8 & n_9. They are without any BTL and then increasing the BTL as shown in Table 5-4, 6.56% (BTL-1), 18.56% (BTL-2) and 37.12 % (BTL-3). The following figures compare the application performance of the use of FRDM routes by splitting packets based on 1:1 and 2:1 ratios. It also shows how applications behave when sending all the packets via the SP.

When comparing only 1:1 and 2:1 distribution ratios used in the FRDM scenario, all statistics show better performance when distributing packets of a flow with 1:1 distribution compared to a 2:1 distribution. The current proposed SF distribution algorithm considers only the PL (section 5.2.2) to decide the distribution ratio. Therefore, in this particular scenario, it calculates the distribution ratio as 2:1 since the BTL in the middle affects the upper path more than the lower path. Although the effect of BTL is higher in the upper path, this path has one hop less than the number of hops in the lower path. When using multi-hop ad hoc networks, the use of each additional hop also introduces an additional delay, which is shown as around 2 ms in this scenario. Therefore, the additional hop delay in the lower path is compensated by the additional effect of BTL in the upper path. Therefore, both the upper and lower paths in the string topology have more or less similar transmission delay. That is why the SF distribution with a 1:1 distribution ratio shows lower delays for audio, video and FTP flows compared to a 2:1 distribution ratio computed by the proposed SF distribution algorithm.

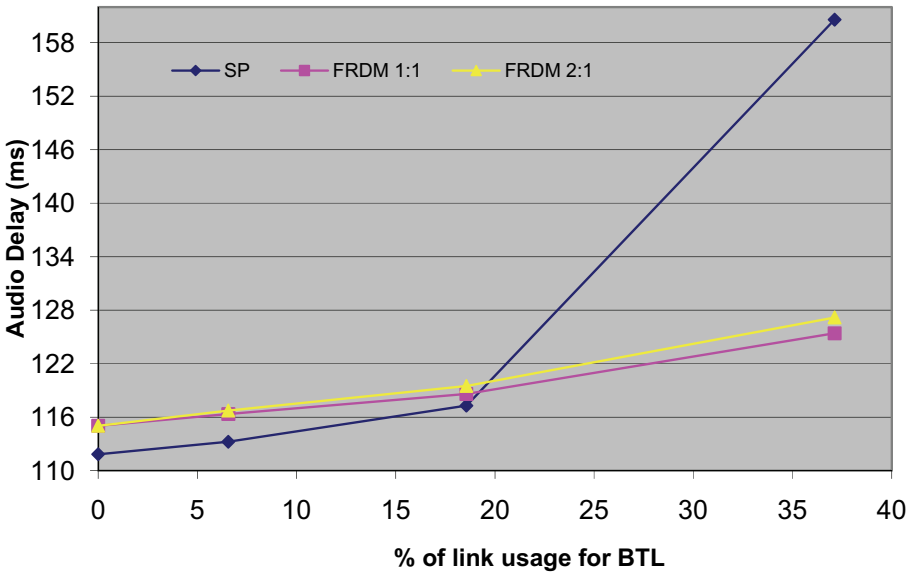


Figure 5-19: String topology – Audio delay

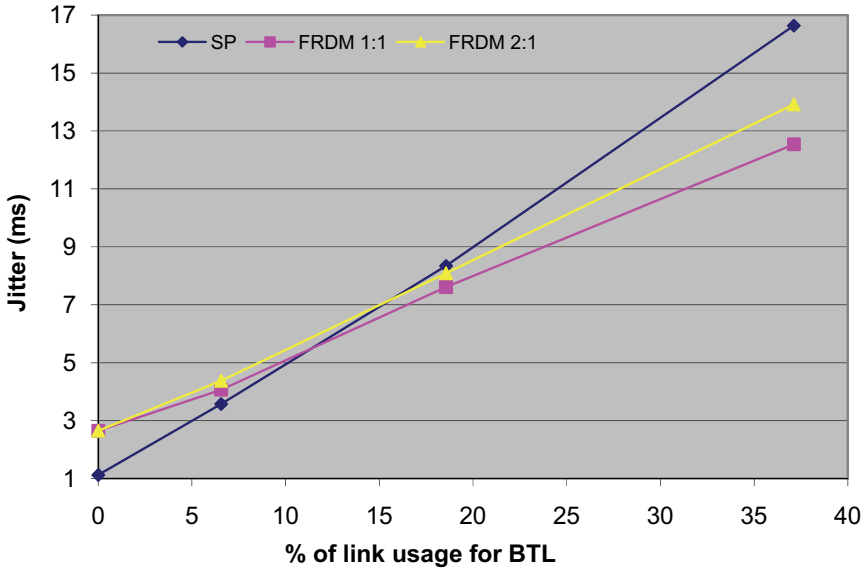


Figure 5-20: String topology – Jitter performance

Comparing FRDM (1:1 distribution) performance with the SP, UDP based flows (both audio and video) in Figure 5-19, Figure 5-20 and Figure 5-21 show the better performance when using the SP with lower BTL (audio until 20% and video until 37% of the use of BTL). However, with the increase of BTL, the SP performance degrades while the performance of FRDM improves as shown in Figure 5-19 to Figure 5-21

Table 5-34 String topology - SF distribution ("audio conferencing")

Scenario	Jitter, ms	Audio Conferencing Delay		WLAN Layer	
		Average, ms	CV	Data Packets dropped	Retransmission attempts
SP (BTL-1)	3.421 ±0.0445	113.14 ±0.0540	0.0388 ±0.0021	0	2148 ±63
FRDM (BTL-1)	3.960 ±0.0421	116.10 ±0.0608	0.0343 ±0.0004	2 ±1	1514 ±30
Perf. Gain	-15.75%	-2.61%			29.51%
SP (BTL-2)	8.393 ±0.0813	117.47 ±0.3012	0.1327 ±0.0795	10 ±2	3963 ±63
FRDM (BTL-2)	7.513 ±0.0569	118.50 ±0.0707	0.0584 ±0.0005	10 ±2	2340 ±40
Perf. Gain	10.48%	-0.87%			40.95%

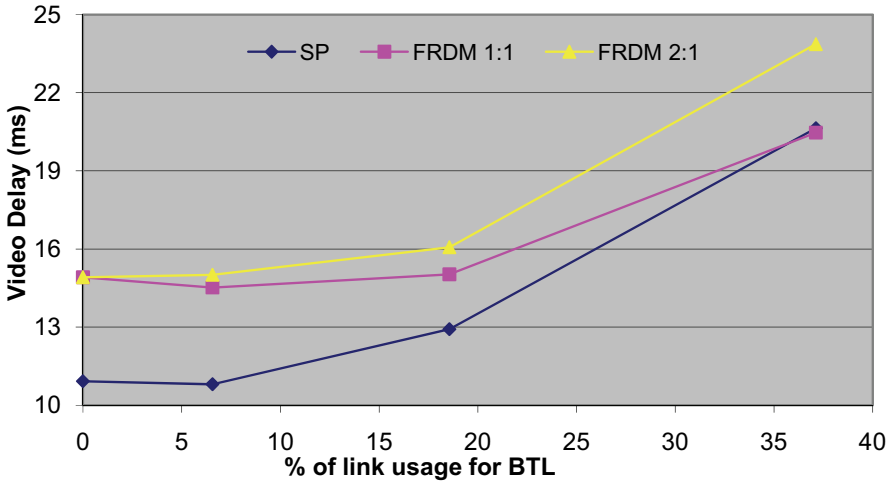


Figure 5-21: String topology – Video delay

Table 5-35 String topology - SF distribution ("video transmission")

Scenario	Video Delay		WLAN layer	
	Average, ms	CV	Data Packets dropped	Retransmission attempts
SP (BTL-1)	11.13 ±0.337	0.227 ±0.0088	0	425 ±277
FRDM (BTL-1)	15.09 ±0.342	0.278 ±0.0130	0	622 ±227
Perf. Gain	-35.57%			-46.35%
SP (BTL-2)	12.88 ±1.558	0.301 ±0.045	0	1631 ±1466
FRDM (BTL-2)	15.96 ±0.762	0.306 ±0.032	0	747 ±305
Perf. Gain	-23.91%			54.19%

As shown in Figure 5-22, in contrast to the performance of UDP flows, FTP DRT is lower when using a lower BTL (until 20% of BTL) for the FRDM scenario. When using

SP, more TCP-Data segments are lost when trying to transmit together with the BTL. But, on the other hand, the performance of FRDM degrades with the increase of the BTL in the SP. This is due to the effect of BTL on both the upper and lower paths. With the increase of BTL, it's difficult for nodes n_5 and n_{15} to access the media to transmit the TCP-Data segments that are received from the node n_D. This happens due to the capturing of the wireless channel by the frequent transmission of UDP packets of the BTL in the middle path.

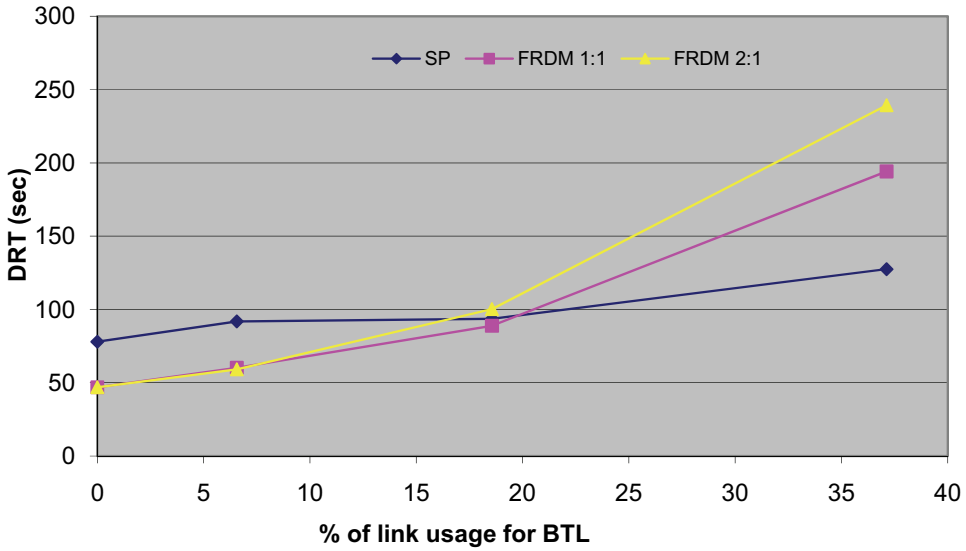


Figure 5-22: String topology – FTP download response time

Table 5-36 String topology - SF distribution ("FTP download")

Scenario	FTP Download		Retransmissions @ TCP	
	DRT, sec	Segment delay, ms	Due to timeouts	Due to DupAcks
SP (BTL-1)	86.58 ±3.86	237.71 ±11.90	18 ±2	20 ±2
FRDM (BTL-1)	61.00 ±5.39	211.40 ±3.21	10 ±3	14 ±2
Perf. Gain	29.54%	11.06%	44.44%	30%
SP (BTL-2)	89.71 ±2.42	267.27 ±12.07	16 ±1	22 ±2
FRDM (BTL-2)	85.78 ±4.03	203.22 ±4.80	21 ±2	21 ±3
Perf. Gain	4.38%	23.91%	-31.25%	4.5%

Table 5-34, Table 5-35 and Table 5-36 show the detailed statistic collected for the audio conferencing, the video transmission and the FTP download, respectively. The explanations given above are valid for the analysis of these results as well. The statistic of number of retransmission attempts in Table 5-34 and Table 5-35 prove the increase

of congestion in both SP and the FRDM scenarios with the increase of the BTL. Table 5-36 shows that there is a higher number of TCP retransmissions done due to TCP timeouts when using FRDM routes with the increase of the BTL. Most of the TCP-Data segments are dropped by the nodes n_5 and n_{15} due to the capturing of channel by the BTL.

5.4.3 Grid Topology

In the grid topology, P_1 and P_3 are selected as the FRDM paths, P_1 and P_2 are selected as the NRDM paths and only P_1 is used in the SP scenario. Though P_1 and P_3 have the same number of hops, they have different PL due to the effect of BTL in P_3 (section 5.1.1.3). The IP packets are distributed to the selected paths in a round robin manner according to the computed *Path Weight*.

Table 5-37 Grid topology- SF distribution ("audio conferencing")

Scenario	Jitter, ms	Audio Conferencing Delay		WLAN Layer	
		Average, ms	CV	Data Packets dropped	Retransmission attempts
SP	2.41 ±0.031	120.66 ±0.053	0.0431 ±0.0004	7 ±6	2665 ±1501
FRDM (dist. 2:1)	2.37 ±0.036	119.28 ±0.039	0.0314 ±0.00032	0	727 ±20
Perf. Gain	1.65%	1.14%			72.72%
NRDM (dist. 2:1)	2.73 ±0.067	120.34 ±0.058	0.0413 ±0.00049	1 ±1	1199 ±33
Perf. Gain	-13.27%	0.26%			55%

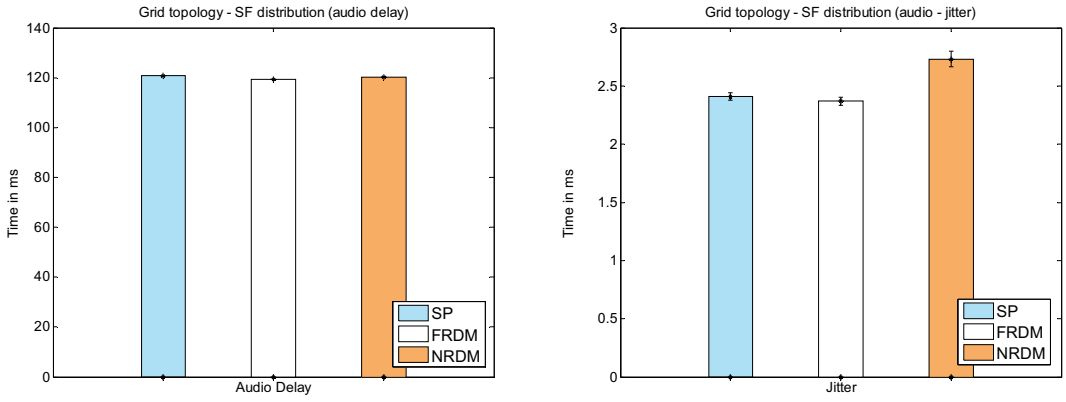


Figure 5-23: Grid topology – SF distribution ("audio conferencing")

Table 5-37 together with Figure 5-23 shows the evaluated parameters when splitting the audio packets over the FRDM routes. It helps to alleviate the load carried by a single path reducing the congestion. Therefore, the FRDM shows a lower audio delay and a lower jitter

compared to both the SP and the NRDM scenarios. A higher number of retransmission attempts done at WLAN, when using the SP route show the congestion in the path. When using the SP, the retransmissions are done by the nodes in SP itself. When using the FRDM routes, one third of the total retransmission attempts are done by the nodes that carry the BTL. Since the audio packets are smaller in size and arrive sooner than the BTL packets, audio packets that are on the secondary path (P_3) are sent sooner. As in the basic topology, the NRDM does not show any promising improvements when splitting the audio flow among interfering routes.

Table 5-38 Grid topology- SF distribution ("video transmission")

Scenario	Video Delay		WLAN Layer	
	Average, ms	CV	Data Packets dropped	Retransmission attempts
SP	17.83 ±0.093	0.219 ±0.0031	0	5 ±2
FRDM (dist. 2:1)	18.48 ±0.187	0.241 ±0.0140	0	478 ±121
Perf. Gain	-3.64%			-94.6%
NRDM (dist. 2:1)	17.87 ±0.122	0.217 ±0.0034	0	83 ±52
Perf. Gain	-0.22%			-15.6%

In contrast to the splitting of the audio packets, Table 5-38 proves that the use of SP is better than splitting the video packets among either the FRDM or the NRDM routes. The audio packets arrive exponentially with very small packet sizes while video packets arrive constantly at a lower rate in different packet sizes. The SP (P_1) is not interfering with the BTL and the packet inter arrival rate of video is not as much as with audio. This reduces the congestion on the SP. When using FRDM routes, the secondary path is congested with the BTL and the video packets that travel via the secondary path causes the performance degradation. Both the packet arrival rate of the BTL and the video flow are also at a constant rate and therefore most of the packets lost when trying to transmit via the secondary path are due to the effect of BTL.

Table 5-39 shows the splitting of an FTP download for different *Path Weight* ratios when using the FRDM routes in the grid topology. It shows the fastest FTP DRT is achieved with the *Path Weight* ratio of 2:1, which is equal to the ratio computed based on the PL. In general, most of the TCP-Data segments are dropped by the nodes n_{16} , n_{17} and n_{18} due to the capturing of the channel by the BTL generated from the node n_6 to node n_7 (TCP-Data segments are sent on the opposite direction to the BTL transmission).

Compared to 1:1 and 2:1 distribution ratios, there are more packets sent over the secondary path with 1:1 ratio. This causes more losses on P_3 with a higher DRT. The

scenario with 20:10 is similar as sending the packets over two single paths, one is on P_1 and the other is on P_3, but, P_3 is also carrying a BTL. The DRT with 20:10 is larger than the DRT in the SP since more TCP-Data segments are lost when sending multiple packets over the secondary path, P_3.

Table 5-39: Grid Topology - SF distribution (FTP download with different *Path Weight* ratio)

Scenario (P_1 : P_3)	DRT, sec	TCP Retransmissions due to		TCP-Data segments dropped by n_{16}, n_{17} and n_{18}
		Timeout	DupAcks	
SP (-)	81.19	14	18	-
FRDM (1:1)	97.17	22	15	25
FRDM (2:1)	65.89	9	16	7
FRDM (20:10)	97.23	19	18	30

The results in Table 5-39 show a bulk distribution rate (FRDM with 20:10 ratio) does not improve the performance since it creates more congestion on one path and behaves similar to the SP scenario. The splitting of audio and video packets with 20:10 distribution ratio also shows the same behavior due to the increase of congestion in one path.

Table 5-40 Grid topology - SF distribution ("FTP download")

Scenario	FTP Download		Retransmissions @ TCP	
	DRT, sec	Segment delay, ms	Due to timeouts	Due to DupAcks
SP	81.47 ±3.31	262.22 ±7.74	13 ±2	19 ±2
FRDM (dist. 2:1)	64.55 ±2.91	252.42 ±5.30	9 ±1	18 ±2
Perf. Gain	20.76%	3.73%	30.76%	5.26
NRDM (dist. 2:1)	135.38 ±10.06	255.32 ±16.63	35 ±3	3 ±1
Perf. Gain	-66.17%	2.72%	-169%	84.21%

Table 5-40 shows the detailed comparison of the SP, the FRDM with 2:1 distribution ratio and NRDM with 2:1 distribution ratio for the FTP download. As in the basic topology, these results prove that the use of interfering routes (NRDM scenario) to split TCP packets does not show any gain in performance improvement.

5.4.4 Random Topology

Table 5-41 together with Figure 5-24 and Figure 5-25 show the statistic collected with the SF distribution in the random topology as explained in section 5.1.1.4. Both the SP and the RDM scenarios are run without BTL and also with a 36% of BTL (denoted as BTL-3 in Table 5-4). The BTL is originated from node n_{15} to node n_{16} in Figure 5-4.

Both the FTP download and the audio conferencing are run independently for each scenario with SF distribution.

Table 5-41 Random Topology - SF distribution ("FTP" and "audio")

Scenario	FTP	Audio Conferencing		
	DRT, sec	Jitter, ms	Delay, ms	CV
SP	56.40 ±3.55	0.723 ±0.062	105.32 ±0.474	0.034 ±0.008
RDM	46.78 ±2.73	2.79 ±0.588	106.78 ±0.559	0.025 ±0.002
Perf. Gain	17.05%	-285%	-1.38%	
SP with BTL-3	313.80 ±57.21	16.68 ±0.702	135.77 ±4.13	0.349 ±0.026
RDM with BTL-3	268.06 ±65.42	15.98 ±1.99	128.11 ±1.862	0.285 ±0.026
Perf. Gain	14.58%	4.2%	5.64%	

The application performance degrades for both the SP and the RDM scenarios when running with the BTL. The comparison of the SP and RDM performance shows a similar behavior as the results taken using the string topology. With the absence of the BTL, the SP is discovered mostly with 3 or 4 hops while the RDM paths are discovered with more hops than the SP. Therefore, when splitting audio packets among the paths which have a higher number of hops does not show any performance gain compared with the SP route. The audio packets propagate faster via the SP due to having a less number of hops. Further the audio jitter is higher when splitting among the RDM paths due to the distribution of packets among paths with non identical hops. However, it shows that there are about 25 audio packets lost when sending all packets via the SP. As in the string topology, the SP performance starts degrading with the increase of the BTL. The SP is discovered including the nodes that carry the BTL. In contrast to the SP, the RDM routes try to discover paths with the nodes that do not carry any BTL. Therefore, the RDM scenario shows better performance with the increase of the BTL.

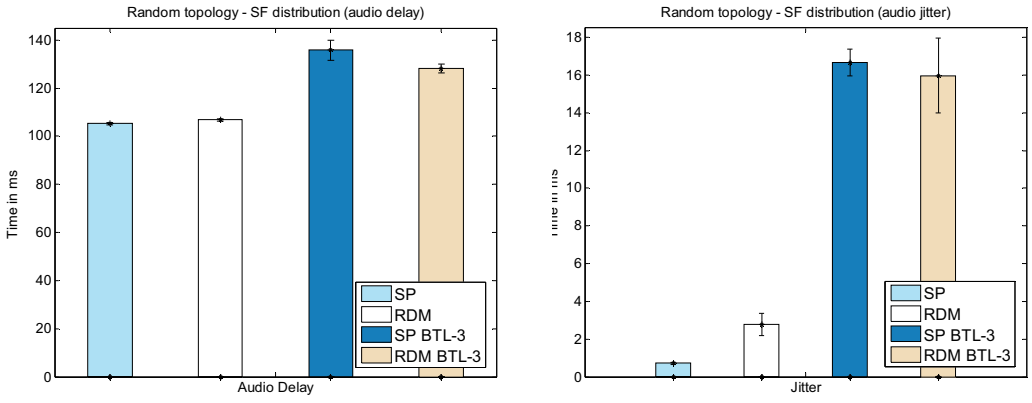


Figure 5-24: Random topology – SF distribution ("audio conferencing")

As explain in Figure 5-22 in section 5.4.2, FTP DRT is faster when splitting TCP-Data segments among RDM paths than on sending multiple TCP-Data segments on the SP. The TCP-Data segments are dropped by the intermediate nodes due to unsuccessful RTS/CTS handshake with the increase of the congestion in the SP. In contrast to the performance of the string topology, FTP in the random topology shows better performance even after increasing the BTL. A further analysis of the routes discovered in the random topology shows that the RDM routes are less affected by the BTL in the middle of the network, compared to the effect of BTL in the string topology. In random topology, the RDM protocol has an opportunity to discover a best pair of paths due to receiving a large number of disjoint paths than in the string topology.

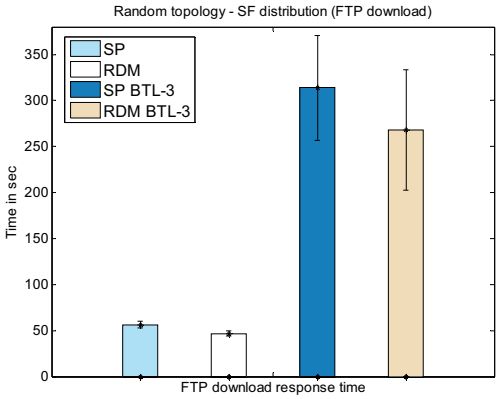


Figure 5-25: Random topology – SF distribution (“FTP download”)

5.4.5 Mobility Scenario

The SF distribution is evaluated using 3 application flows (i.e. FTP, audio and video) in the mobile scenario. All the parameters evaluated for the MF distribution are also considered here. In addition to those parameters, Packet Delivery Ratio (PDR), which gives the percentage of total number of packets received w.r.t. total number of packets sent are computed.

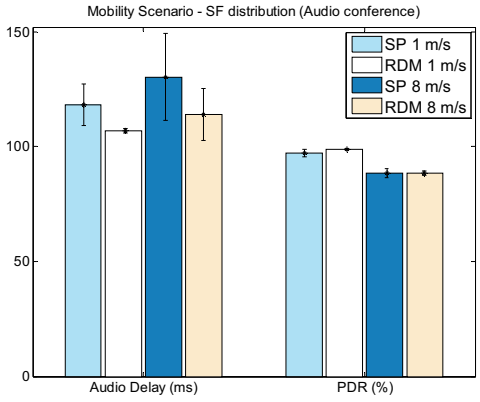


Figure 5-26: Mobility Scenario – SF distribution (“audio conferencing”)

Table 5-42 together with Figure 5-26 shows the evaluated parameters when splitting audio packets among RDM routes and sending all the packets on one path (SP).

Table 5-42 Mobility scenario - SF distribution ("audio conferencing")

Scenario	Jitter, ms	Delay, ms	CV	PDR, %
SP –1 m/s	1.03 ±0.252	118.16 ±9.101	1.26 ±0.834	97.09 ±1.598
RDM –1 m/s	1.77 ±0.444	106.98 ±0.999	0.109 ±0.061	98.939 ±0.335
SP –8 m/s	1.62 ±0.306	130.26 ±18.982	1.83 ±0.804	88.34 ±1.961
RDM –8 m/s	1.78 ±0.146	114.07 ±11.394	0.795 ±0.625	88.47 ±0.996

With the increase of mobility, the performance starts degrading with a higher number of packet losses for both the RDM and the SP scenarios. But, compared to the use of SP, the splitting among RDM paths shows better performance for all the cases. Compared to the outcome of the random topology without any BTL, SP does not perform better with the mobility. The main reason for this is explained by analyzing the number of route discoveries generated in the SP and the RDM paths (see Table 5-43).

Table 5-43 Mobility scenario - SF distribution ("audio conferencing") – route discovery parameters

Scenario	RREQs sent	Successful Route Discoveries		RERRs sent	Average RDT, ms		
SP –1 m/s	14 ±6	14 ±6		62 ±30	230.22 ±106.48		
SP –8 m/s	46 ±8	43 ±7		219 ±57	667.75 ±400.05		
						% of usage of routes	
		SP	RDM			SP	RDM
RDM –1 m/s	4 ±1	0 ±1	3 ±1	23 ±10	152.17 ±9.38	54 ±14	46 ±14
RDM –8 m/s	26 ±3	7 ±3	18 ±3	218 ±46	234.93 ±80.19	66 ±5	34 ±5

Table 5-44 Mobility scenario - SF distribution (video & FTP download)

	Video Transmission			FTP Download	
Scenario	Delay, ms	CV	PDR, %	DRT, sec	Segment Delay, ms
SP –1 m/s	10.75 ±7.824	5.82 ±1.565	95.83 ±3.39	57.26 ±4.79	192.49 ±20.42
RDM –1 m/s	8.02 ±1.36	3.21 ±1.201	98.13 ±0.498	52.28 ±10.59	235.10 ±51.03
SP –8 m/s	21.92 ±9.053	9.72 ±1.021	87.07 ±1.978	60.17 ±9.76	200.36 ±20.04
RDM –8 m/s	12.29 ±7.265	9.25 ±0.675	87.88 ±1.156	56.85 ±7.35	237.80 ±40.46

The RDM protocol generates fewer route discoveries since it initiates route discoveries after the expiry of both of the RDM paths. Table 5-44 together with Figure 5-27 shows how the video conferencing and the FTP download behave in the mobile scenario. As in audio performance, both the video and the FTP download also show better performance when splitting packets among RDM routes. Table 5-45 and Table 5-46 show all the parameters used to evaluate the route discovery process of the SP and the RDM protocol.

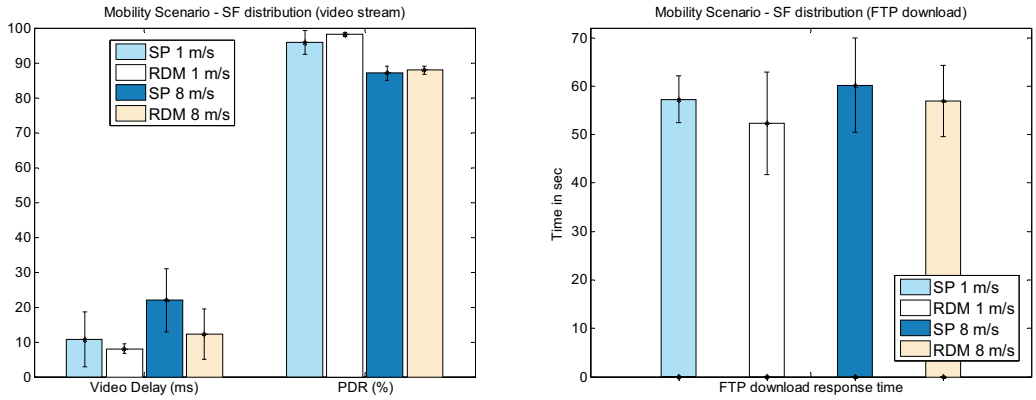


Figure 5-27: Mobility Scenario – SF distribution ("video transmission" and "FTP download")

Table 5-45 Mobility scenario - SF distribution ("video transmission") – route discovery parameters

Scenario	RREQs sent	Successful Route Discoveries		RERRs sent	Average RDT, ms		
SP –1 m/s	13 ±9	13 ±9		50 ±45	255.82 ±132.42		
SP –8 m/s	51 ±8	47 ±7		187 ±41	377.92 ±75.55		
						% of usage of routes	
		SP	RDM			SP	RDM
RDM –1 m/s	5 ±1	1 ±0	4 ±1	30 ±9	159.10 ±13.29	52 ±12	48 ±12
RDM –8 m/s	26 ±4	5 ±2	19 ±2	159 ±27	185.43 ±27.27	63 ±6	37 ±6

Table 5-46 Mobility scenario - SF distribution ("FTP Download") – route discovery parameters

Scenario	RREQs sent	Successful Route Discoveries		RERRs sent	Average RDT, ms		
SP –1 m/s	3 ±1	3 ±1		5 ±3	97.07 ±14.79		
SP –8 m/s	5 ±2	4 ±2		11 ±6	116.76 ±26.71		
						% of usage of routes	
		SP	RDM			SP	RDM
RDM –1 m/s	2 ±0	0	2 ±0	0	157.59 ±15.09	31 ±10	69 ±10
RDM –8 m/s	3 ±1	1 ±1	3 ±1	8 ±4	146.48 ±9.05	58 ±9	42 ±9

Table 5-45 and Table 5-46 conclude the following.

- There is less number of route discoveries, RERRs and RREQs generated when running with the RDM protocol irrespective of the type of mobility used.
- With the increase of mobility, the RDM protocol tends to discover more SP routes and also the percentage of usage of SP routes are increasing for all 3 application flows.

5.5 Conclusion

This chapter discusses the simulation environments, different types of network topologies used, application flows used and the proposed MF and SF distribution algorithms in detail. It further provides a detailed analysis of the results taken using different scenarios and compares the performance of the simultaneous use of RDM routes together with the standard SP route. The evaluation of the results of different distribution algorithms that are used to distribute flows among RDM routes concludes the following.

MF Distribution: The MF distribution is implemented based on the *worst fit algorithm*, a solution of a well known computer allocation problem known as *Bin packing problem*. This distributes the traffic load over 2 RDM paths according to the remaining bandwidth. The remaining bandwidth is computed based on the PL, which itself depends on the mutual interference between two paths and the BTL available on each path.

- Simulation results show there is a significant improvement when using FRDM routes with MF distribution compared with the performance on the SP. This is mainly due to alleviating the congestion on one path by distributing individual flows among non-interfering paths. The use of SUM routing with MF distribution in mobile scenarios show much better performance compared to the SP. This is mainly due to the use of 2 paths which causes less route discoveries and also alleviating the congestion on one path when both paths are active. The percentage of performance improvement varies depending on the topology used. In summary, video delay is decreased between 31% ~ 60%, audio jitter is decreased between 13% ~ 49%, FTP DRT is decreased between 23% ~ 45% and HTTP ORT and PRT is decreased between 12% ~ 29%, when using the FRDM routes simultaneously for all the investigated topologies.
- The use of interfering routes simultaneously (NRDM) degrades the performance especially for TCP traffic. When transmitting TCP traffic on one path while the other path is carrying UDP traffic, the nodes carrying the UDP traffic capture the channel. Therefore, the nodes carrying TCP traffic have to wait too long and finally dropping the TCP packets due to the capture of channel by the other path. This does not happen when using the FRDM routes simultaneously since both paths are not interfering with each other.

SF Distribution: The SF distribution uses the round robin distribution to split packets. The distribution ratio is computed based on the PL, forwarding more packets to the path with a lower PL. The SF distribution is analyzed mainly for a single flow. The performance of the use of RDM routes with the SF distribution (when a single flow is present) in the investigated topologies concludes the following.

- *Distribution among identical paths:* When the RDM paths have identical properties (PL and hop counts), all application flows (audio, video, FTP & HTTP) investigated outperform the SP route. The splitting of packets among identical RDM paths reduces the congestion to half on each path compared to forwarding all packets on one path. The use of the FRDM with SF distribution shows performance improvement for both UDP based and TCP based traffic, while the NRDM shows a marginal performance improvement only for UDP based traffic. When splitting TCP packets among NRDM paths, TCP packets are lost due to the capturing of the channel by the nodes in the other interfering paths. The capturing of channel affects badly for asymmetric communications like TCP.
- *Distribution among non identical paths:* The performance when splitting traffic among non identical paths differs depending on type (UDP or TCP based) of traffic. Splitting of UDP traffic performs better if the traffic on the SP creates more congestion. In this case, splitting UDP packets even among non identical paths show an improvement due to alleviating the congestion (splitting of the audio flow in the Grid topology). When the SP has shorter hops than the RDM paths, SP performs better even with BTL up to 20% in the investigated scenario (string topology). But, with the increase of BTL, splitting the packets among RDM paths starts performing better than sending all the packets over the congested SP. As in other scenarios, splitting TCP traffic among FRDM paths shows better performance, while splitting among NRDM shows a degrading performance. Further, a bulk distribution rate (20:10 ratio in the grid topology) does not improve the performance since it creates more congestion on one path and behaves similar to the SP scenario.
- *Mobile Scenarios:* Even though the paths discovered in the mobile scenarios do not have identical properties, the splitting of both TCP based and UDP based packets show an improvement in the performance.

In this thesis, the distribution ratio for the SF distribution is decided based only on the PL. The results of the string topology show that the optimal ratio can be achieved by considering the total transmission delay of selected paths. The transmission delay is affected by the PL as well as the number of hops. Therefore, the computation of the optimal distribution ratio should be done considering the details such as delays due to the BTL, delays due to number of hops, etc. This can be achieved similarly to the approach used in [80] by deciding the distribution criteria based on different context that change dynamically.

SF distribution when multiple flows are present: Splitting of packets of a single flow is also evaluated when multiple flows are present. These results are compared together with the results taken when using the MF distribution to distribute individual flows.

Splitting of both TCP and UDP packets together results in a higher variation in UDP parameters (delay and jitter) and a higher number of data packets to be dropped at the WLAN layer. On the other hand, TCP flows show an improved performance, compared with separating UDP and TCP flows. This is due to a higher probability of accessing the channel (especially when accessing the end nodes in the investigated scenarios) while transmitting together with UDP packets.

Standard SP vs SP discovered by RDM: The performance over the standard SP is also compared together with the SP that is discovered by the RDM protocol. With the increase of the BTL on the standard SP, results show that the SP path discovered by the RDM protocol performs better in terms of jitter variation (reduced by 27%) and the number of packets dropped (reduced by 36%). The RDM protocol selects a path which has the least interference and the least congestion as its SP.

6. Performance Evaluation of RDM Routing: Replicating

This chapter discusses the applicability of RDM routing for emergency scenarios. For example, a more reliable means of data transmission is required in a fire-fighting scenario due to extreme environmental conditions (fire, smoke and vapor). In contrast to balancing the load among RDM routing, replicating each data packet among RDM paths can increase the reliability of the communication. Although replication causes higher congestion in the network, it enhances the reliability, especially in adverse propagation environments such as in fire-fighting. In the next section, an applicable scenario for fire-fighting is introduced summarizing the feasibility studies done to prove the suitability of using wireless technologies for fire-fighting applications. The second section discusses how TCP and UDP based flows react when replicating packets at the sender and receiving redundant copies at the receiver. The third section details the simulation results taken to evaluate fire-fighting applications and the last section concludes this chapter.

6.1 Deployment of MANET in Fire-fighting Scenarios

A particularly challenging emergency scenario with respect to communications is fire-fighting. Coordination and collaboration is crucial to reduce risks and to increase efficiency of fire fighters. Currently, fire fighters often still use conventional analogue radios which are not useful for the transmission of additional status information (e.g., image of a thermal camera, position of fire fighters or still images). The digital communications of TETRA and TETRApol [81] are being used in different countries in Europe. However, these digital radios are not widely deployed due to its cost. When using wireless ad hoc networks, the range of single hop communication may not be sufficient in certain environments, e.g., tall buildings with several floors or inside a tunnel/subway [82]. The use of modern digital communication systems with multi-hop wireless technologies can be used to address some of the above mentioned issues.

In the wearIT@work project funded by the European Commission in the 6th Framework Programme [83], wearable computing for the mobile worker has been evaluated in four application scenarios: aircraft maintenance, hospital care, car production, and fire-fighting (emergency response). In all of these application scenarios, work processes, applications, and technologies for mobile workers have been investigated in detail [2, 84-88].

In summary, the requirements of the fire fighters are identified as follows.

- A fire fighter needs to communicate to the incident commander at the command post and also with the other fire fighters in the vicinity by voice.
- Additional exchange status information (room checked, position, vital data like heart beat) is also important same as the transmission of high quality still images, e.g. of an engine, a chemical container etc.
- A fire fighter needs to get assistance in some environments such as places with chemicals to identify chemical containers or switch off dangerous equipment. This requires downloading files from the command post which gives annotated photos, voice commands or map details.
- Video transmission is mainly interesting for thermal camera images to be transmitted from one fire fighter equipped with the camera to other colleagues in close vicinity, who do not use a thermal camera for cost reasons.

In accordance to these requirements, there should be an advanced communication system to enhance the work process of fire fighters. As fire fighters cannot rely on a specific infrastructure being in place at the place of the emergency, the approach taken is to use any existing communication system backed up by wireless multi-hop communications.

6.1.1 Usage Scenario

MANETs are considered as a candidate for providing flexible and robust communication environments in emergency scenarios like fire-fighting.

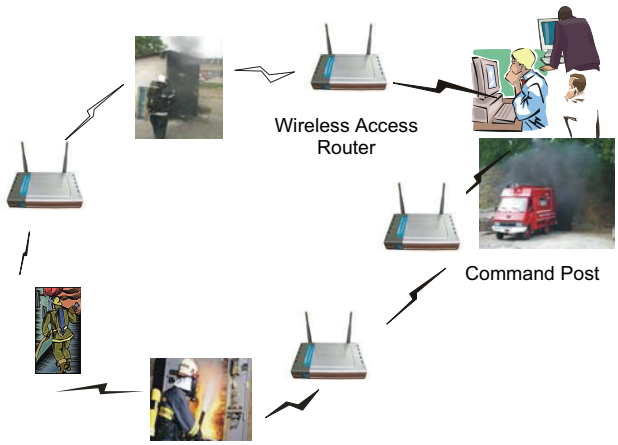


Figure 6-1 Deployment of multipath routing for fire-fighting scenario

Usually fire fighters operate in groups and one group fights the fire for a maximum of about 45 to 60 minutes. Then, another group takes over the activities. Usually, one group consists of 5-10 fire fighters. In such a group based scenario, it is likely that it will be difficult to find multiple routing paths between a fire fighter and the command post. In order to find multiple paths and to maintain reliable connectivity, the fire fighters might place wireless access routers/repeaters at different places which are used

for multi-hop communications. This deployment must of course be included in the normal workflow. Therefore one can imagine that the wireless routers could be integrated into the equipment, the fire fighter uses and places throughout the building, such as pumps, water hoses, toolboxes etc. As shown in Figure 6-1, wireless access routers act as fixed/stationary intermediate nodes through which the fire fighters' devices discover communication paths. The research related to making existing wireless access routers more robust and reliable in these adverse environmental conditions is also considered within this project.

6.1.2 Wireless Technology for Fire-fighting

Several experiments were conducted, with the involvement of the author of this thesis in the wearIT@work project, at the training facilities of the Paris fire brigade (BSPP) and several practical measurements have been taken in a multi-hop ad hoc environment to investigate the feasibility of using existing wireless technologies and devices in adverse propagation environments. The main goal was to conduct several feasibility studies to evaluate what sort of applications can be run in multi-hop ad hoc networks, mainly considering the requirements of fire-fighting applications. This section discusses the outcome of these results in brief.

6.1.2.1 Wireless Propagation Test at BSPP

The experiments have been carried out in a tunnel system at the training facilities of the Paris fire brigade in Villeneuve St Georges, located close to Paris. Only commercial-off-the-shelf components with standard wireless communication technologies (IEEE 802.11a, b and n, Bluetooth and 802.15.4) have been used. Throughput, jitter and Round Trip Time (RTT) have been measured on the network layer to evaluate the performance. The tunnel, which is about 1.5 m wide, 2 m high, 50 m long, made of stone and covered with earth and grass, was used to conduct the experiments. The transmitting device has been located in a stationary position and receiving device was used as the mobile device, while a fire was ignited between the transmitter and the receiver. Analysis of the results concludes the following.

- Technologies using the 2.4 GHz frequency band generally perform much better than technologies using the 5 GHz band in the tunnel system. IEEE 802.11a only achieves a communication range of approximately 25 m inside the tunnel due to the shorter communication range compared to 2.4 GHz technologies.
- Fire and smoke do not affect the communication performance of devices operating in the 2.4 GHz band.
- Vapor reduces the transmission quality by decreasing throughput and range, and increasing jitter.

Although vapor decreases the transmission range by about 20%, communication was still possible at a distance of 40 m, which is considered to be sufficient for the application scenario. On the other hand, the jitter is increased significantly close to the maximum communication range, which can be problematic for voice applications. An

application receiving voice packets can cope with high jitter in two ways, by either dropping the received packets with high jitter or buffering them for a certain time before decoding. More details about this test can be found in [89].

6.1.2.2 Performance of Multi-hop Ad hoc Networks

TCP and voice performance was also evaluated in a stationary IEEE 802.11b multi-hop network with seven hops. The experimental results are compared with the simulation results. The throughput of UDP and TCP decreased with approximately 1/(hop count) for the first three hops and then stays stable for the remaining four hops. The throughput was much higher when transmitting large packets. The average delay and jitter of UDP packets were very low.

The throughput of TCP was only slightly lower than that of UDP. This is acceptable considering that TCP is supposed to perform badly over multi-hop wireless links. The simulation yields the same qualitative and similar quantitative results as the experiments. Finally, voice communication of very good quality was also possible over up to seven hops. More details about this test can be found in [21]. These results clearly indicate that multi-hop communication can satisfy the need for voice communications with good or acceptable delay and jitter. It is also capable of sending the status, vital data (fire-fighter's heart beat, temperature, etc.) and performing still image transfers for remote support.

6.1.3 Research Issues – Deployment of Multiple Paths

This thesis focuses on two aspects of research, concerning the fire-fighting applications.

- Different available wireless technologies and devices were tested for their robustness in adverse propagation environments as discussed in section 6.1.2. These results show that currently available IEEE 802.11 based technologies are suitable to be used in wireless multi-hop ad hoc networks.
- The second area of research consists on determining how to increase the robustness of wireless ad hoc communications by utilizing more than one path simultaneously. The focus here is to deploy the use of RDM paths simultaneously to improve the reliability of the communication. In this thesis, a mechanism of packet replication among RDM routes was proposed to improve the reliability of communication for fire-fighting applications. As shown in Figure 6-1, the RDM paths are discovered by placing fixed wireless ARs in addition to the mobile devices used by the fire fighters.

6.2 Issues in Replicating Packets

This section discusses issues to be addressed when receiving redundant copies of the packets if packets are replicated at the sender. As explained in section 3.3.3.3, both replication and discarding of replicated copies are implemented as configurable options in the OPNET simulator. How different applications are affected when replicating without discarding redundant packets at the receiver are analyzed in detail for both TCP and UDP based applications.

A basic topology which has two identical routing paths between the sender and the destination is selected as shown in Figure 6-2. The source node, S initiates sending data to the destination node, D. When using RDM routes, the packets are replicated and sent via both the upper and lower paths. When using the SP route, packets are sent only via the upper path without replicating.

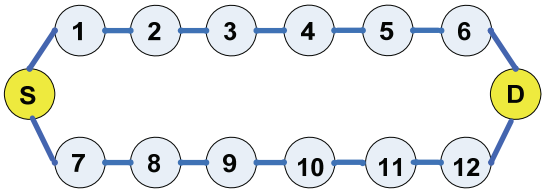


Figure 6-2 Basic topology to evaluate replicating data

A single FTP download is started to download a 1 MB file from the node D. The TCP Reno version is used here with fast retransmission enabled. The FTP download response time is shown in Table 6-1. A shorter response time is taken when redundant packets are discarded at the receiver ends. The highest download response time is shown when using the SP route. This is due to a higher number of retransmission timeouts that occur when using the SP route. When comparing with replicating packets among RDM routes, a loss of a TCP-Data packet sent via one path can be recovered from the replicated copy that comes via the other path. Therefore, there are only a few retransmissions due to timeouts with RDM scenarios than the SP scenario.

Table 6-1 FTP download response time – use of RDM routes with replicating data vs SP route

Scenarios		DRT (sec)	# of retransmissions due to DupAcks	# of retransmissions due to Timeouts
RDM – Replicating (without discarding)	DupAck = 3	64.62	49	-
	DupAck = 6	62.66	14	1
RDM – Replicating (with discarding)	DupAck = 3	55.56	10	1
SP (without replicating)	DupAck = 3	82.24	16	16

Table 6-1 shows that there are more TCP retransmissions due to the receipt of DupAcks, when forwarding the replicated copies to the transport layer. The reasons for triggering more retransmissions with redundant packets are explained in Figure 6-3.

As shown in Figure 6-3, the TCP receiver acknowledges each successfully received packet, regardless of whether it is the original or the replicated packet. Therefore, it generates two TCP-ACKs for a pair of original and replicated TCP-Data packets. Each TCP-ACK is again replicated at the IP layer of the receiver before being sent. Subsequently the TCP sender receives four identical TCP-ACKs for one TCP-Data

packet. Upon receipt of four TCP-ACKs, the TCP sender triggers the fast retransmission since Dup-ACK threshold is set to 3. As shown in Table 6-1, the number of retransmissions can be reduced significantly with the increase of DupAck threshold to 6. On the other hand, the use of higher Dup-ACK threshold does not help to reduce the download response time as the TCP sender has to notify the packet loss with the expiry of the TCP timeout.

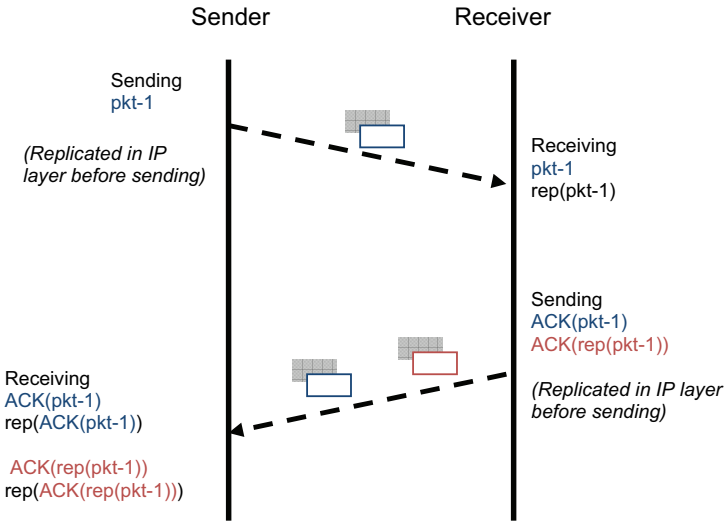


Figure 6-3 TCP-Data and TCP-ACK replication at the sender & the receiver

However, if redundant packets are discarded in the IP layer, there is only one ACK generated for each pair of original and replicated packets and again at the sender, the redundant ACK is discarded. This does not trigger the fast retransmission unnecessarily, when receiving replicated copies of TCP-Data packets. That way, the retransmissions are not falsely triggered.

Table 6-2 Video and audio performance – replicating

Scenarios	Video Delay (ms)	Audio Delay (ms)	Audio jitter (ms)
RDM – Replicating (without discarding)	22.07 ±7.16	122.58 ±7.28	5.795 ±5.012
RDM – Replicating (with discarding)	18.77 ±5.40	119.84 ±5.16	3.592 ±3.534
SP	15.32 ±3.26	117.92 ±4.46	0.039 ±3.906

The standard TCP RFC 2001 [58] does not specify how it handles replicated packets. But [90], which is referred by RFC 2001, has discussed this behavior of TCP. Both out-of-order packets and packet replications (by either the network or the sender) can produce replicated packets, and TCP creates an ACK for each packet received. Therefore, more TCP retransmissions can be falsely triggered. This is the major reason of the poorer performance in TCP based applications when packets are replicated. However, [91] [92] [93] have made further extensions to standard TCP behavior to avoid unnecessary loss recovery that can occur due to packet replications.

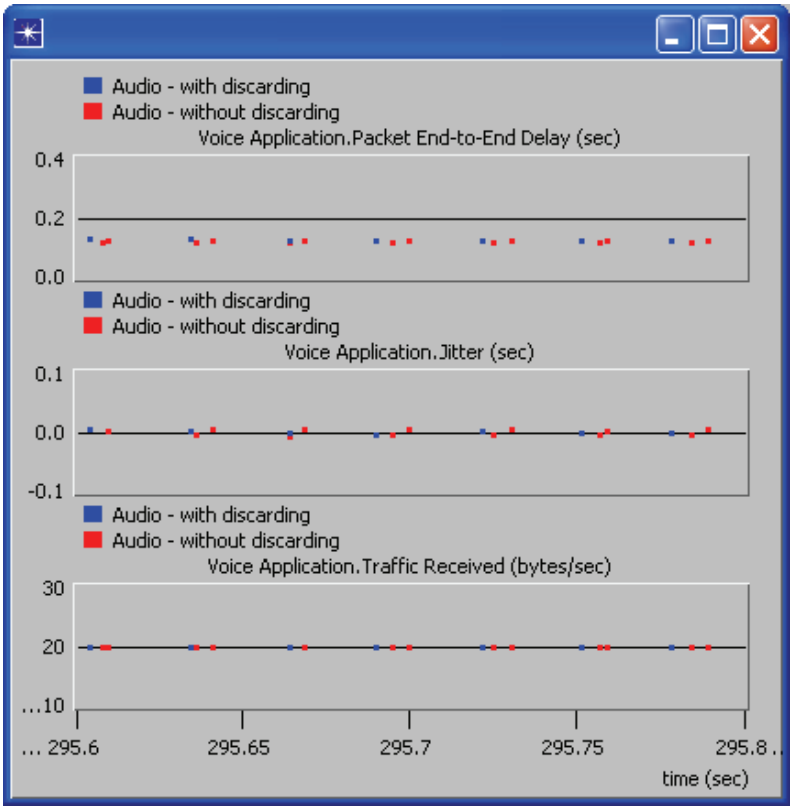


Figure 6-4 Audio packets received with and without discarding

Video transmission and audio conferencing applications are used to evaluate the performance of UDP for both cases of with and without discarding redundant packets. Table 6-2 shows the performance of these applications showing higher delays when forwarding the redundant copies to the application layer. The SP shows the best performance compared with the use of RDM paths with replicating. Replication of packets causes more contention at the end nodes when receiving and transmitting packets though the RDM paths are not interfering with each other. Figure 6-4 compares the number of audio packets received when discarding replicated packets at the IP layer vs when not discarding the replicated packets. It shows that the number of packets that go to the upper layers are doubled when redundant packets are not discarded at the receiver. In UDP based flows, all redundant copies received are forwarded to the application layer, while the redundant copies of TCP-Data are dropped at the transport layer.

6.3 Evaluation of RDM Routing (Replicating)

The replication of packets is implemented in the RDM protocol as explained in section 3.3.3.3. Packet replication enables the source (sender) to replicate each outgoing packet

as many times as the number of entries in its routing table, which are discovered by the RDM protocol. The implementation of replicating and splitting the original and the replicated packets among the RDM routes are done at the IP layer since the number of active routes available can be extracted at this layer. The identification of replicated packets and then to discard the redundant packets are implemented at the receiver. The details of discarding of redundant packets can be found in Figure 9-3 in Appendix II. This section analyses the results taken using both the stationary and the mobile scenarios with replicating packets and discarding of redundant packets.

6.3.1 Evaluation of RDM routing with replication in stationary networks

Figure 6-5 is set up to evaluate the behavior of replicating when using 3 different types of RDM routing. The wireless nodes used in all the scenarios use the same properties as explained in Table 5-1.

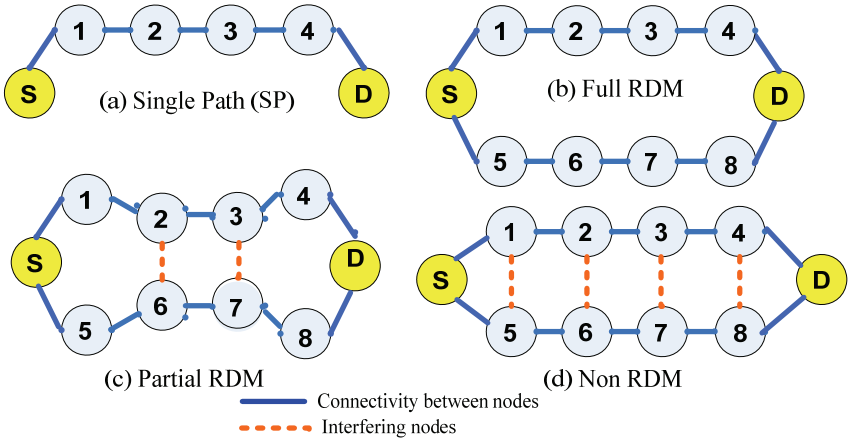


Figure 6-5 SP vs 3 types of RDM routing to replicate data – Scenario 1

Scenario-1 As shown in Figure 6-5, they are a full RDM (where all intermediate nodes of each path are not within the same WLAN interference region) network, a partial RDM (intermediate nodes in the middle are interfering to each other) network and a non-RDM (where all intermediate nodes are within each other's interference regions) network. The performance of these networks is compared against single path (SP) routing. When using RDM paths, packets are replicated at the IP layer before sending and the redundant copies are discarded at the IP layer upon the receipt at the destination. When using the SP routing (Figure 6-5 a), all packets are sent over one path without replicating. A unidirectional UDP stream is sent from S to D and the maximum throughput is measured at the application layer. The maximum possible throughput (sustainable throughput) is found by identifying the throughput until there is a data loss, experienced at the link layer (due to the contention, interference, buffer overloads, etc). Further, the sustainable throughput that can be achieved as a percentage of the PHY mode (1 Mbps) throughput is used to compare the results.

Table 6-3 Sustainable throughput (%) – Scenario 1

Types of Routing	%
Single Path – SP	25.10
Multiple Path - FRDM	22.40
Multiple Path - PRDM	16.00
Multiple Path - NRDM	10.40

Table 6-3 shows that the sustainable throughput of SP is higher than the simultaneous use of RDM routing with replicating data. Compared to non-RDM and partial RDM, full RDM routing still shows comparable improvements in throughput. In scenario 1, the sustainable throughput of full RDM routes is lower than the SP routing due to the congestion created at the end nodes, while accessing the S and the D from both the upper and lower paths simultaneously. In summary, replication causes higher congestion in the network. This effect is higher when using applications with higher data rates and simultaneous use of paths which are interfering with each other. As investigated earlier, most of the applications used for fire-fighting do not require a high bandwidth, except for the video transmission (up to 200kbps is enough even using 1 Mbps of PHY mode) and more importantly, the use of full RDM routes enhances reliability, which is proven with results shown subsequently.

6.3.2 Evaluation of RDM routing with replicating in mobile networks

Scenario-2 (Figure 5-5) **and Scenario-3** (Figure 6-6) show the use of the RDM protocol in fire-fighting applications under simulated mobility. The mobility in both scenarios is configured using the Random Waypoint Mobility (RWM) model with the speed of 1 m/s without any pause time. Scenario 2 is configured using a RWM model for a network consisting of 30 mobile nodes (representing fire fighter`s devices). The nodes are deployed randomly in a 1.8 km x 2.4 km area (Figure 5-5). Scenario 3 uses wireless ARs to extend the multi-hop communications and are configured to operate as fixed (stationary) nodes. Here, the mobility is defined for 3 groups of firefighters (FF-1, FF-2 to FF-6 and FF-7 to FF-10) by restricting the mobility area as shown in Figure 6-6. Both scenarios are run for 15 minute durations. The results are taken for 10 different seeds and are shown with 95% confidence interval. Measurements are repeated for both SP and RDM path routing. With RDM routing, each data packet is replicated at the IP layer at the sender and the redundant packets are discarded at the receiver at the same layer, when using more than one path. The RDM routes are configured to find full RDM paths to use non-interfering paths simultaneously. The following applications are used to compare the performance of RDM routes and SP routing.

- *Unidirectional UDP flow:* Periodic retrieval of sensor data at each second from FF_1 to the *Command Post*. Size of UDP packet is 125 bytes.
- *Bidirectional UDP flow:* Audio Conferencing between the FF-1 and the *Command Post*. A VoIP audio conferencing flow is configured with G 723.1 codec as explained in section 5.1.2.

- *TCP flow:* multiple FTP downloads are used to simulate the sending of data files (e.g. with annotated photos, with map details) from the *Command Post* to the FF_1. The size of a file is set to 100000 bytes and sent in 30 seconds intervals. The multiple FTP downloads are active for 700 seconds duration.

Table 6-4 Packet Delivery Ratio (PDR) – Scenario 2

FF Applications	PDR (%)		
	SP	RDM - Replicating	RDM - Splitting
Sensor Data	94.03 ±6.32	98.83 ±0.74	96.55 ±1.29
Audio conferencing	93.61 ±8.27	99.04 ±0.64	96.59 ±1.87

Scenario 2 is configured to run only with UDP based applications, while scenario 3 is configured with the 3 applications mentioned above. Table 6-4 shows that the use of full RDM routes in Scenario 2 increases the PDR for both UDP based applications. The PDR is computed as the percentage of total packets received w.r.t. the total number of packets transmitted.

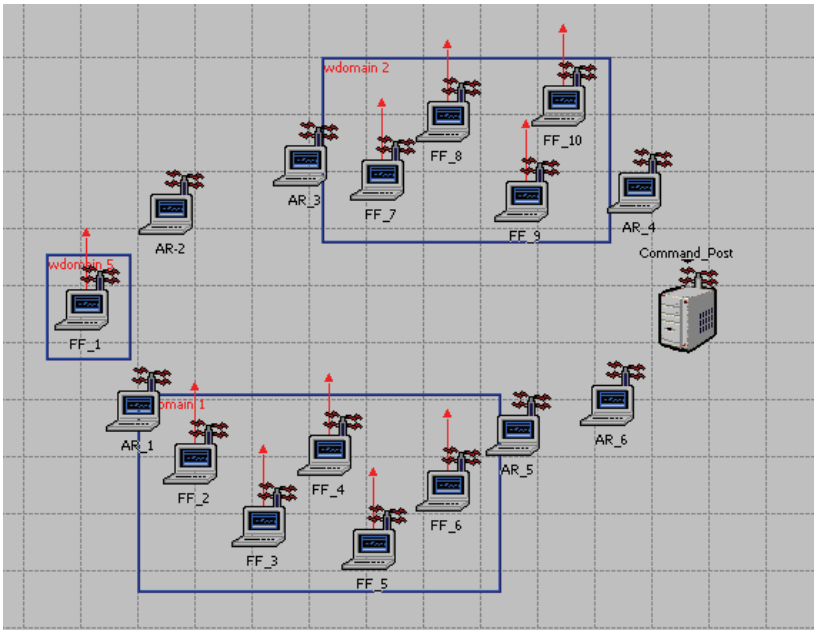


Figure 6-6 Use of RDM routing (with random mobility) – Scenario 3

Scenario 2 with RDM is run with and without replicating packets. RDM-Replicating in Table 6-4 means 2 paths are discovered and used simultaneously where each UDP packet is replicated and distributed among 2 non-interfering paths. In case of RDM-Splitting, UDP packets are distributed (in a round robin distribution) among both RDM paths without replicating. In SP routing; only a path with the lowest hop count is used since it contains fewer hops and UDP packets of two applications are sent over one

path.

Table 6-4 shows SP routing has the lowest PDR, while the RDM-replicating shows the highest PDR. The movement of firefighters during the communication causes some packets to be lost during the route re-discoveries. In general, SP shows higher route discoveries than the RDM routing. A break of one path does not initiate another route discovery since the packets are forwarded over the other active path when using RDM routing. The PDR of RDM-Replicating is higher than the RDM-Splitting, since the packets that are lost on the broken paths have been recovered from the replicated packets that come over the other active path. Though the packets lost due to environmental conditions are not simulated here, these results show that the probability of recovery of lost packets is higher with RDM routing due to the transmission of replicated copies.

Table 6-5 shows the percentage of total traffic received w.r.t. the total number of traffic transmitted in scenario 3. Most of the lost packets during the breakage of routes due to mobility and also due to contention are recovered in RDM, from the replicated data which arrives via the non broken path as in scenario 2. Therefore, the percentage of packet losses is lower when using RDM routing and replicating data. Table 6-6 shows that the number of route discoveries ("Total number of RREQs attempts") is much lower when using RDM paths. When having 2 paths and if one path fails, the RDM protocol does not initiates a route discovery and data is sent over the other active path without replicating. Therefore, there is a possibility of using only one path without replicating data when using the RDM protocol. Table 6-6 shows that 52% of the simulation time, the RDM protocol also uses the SP routing.

Table 6-5 Packet Delivery Ratio (PDR) – Scenario 3

FF Applications	PDR %	
	SP	RDM
Sensor Data	92.97 ±3.49	96.90 ±1.19
Audio conferencing	94.52 ±2.79	97.45 ±1.04
Completed number of downloaded files	89.89 ±9.28	94.17 ±2.76

It is also observed that the initiation phase of the FTP download may not be successful during the route re-discoveries due to the following reason.

- After a route break, it requires more time to find the route again. During this time, the originating node (either FF-1 or Command Post) starts buffering the data. These data are released immediately once the routes are found. This creates a higher contention at the WLAN since it consists of all audio, UDP and TCP packets. In case of a loss of a TCP control packet (SYN, ACK-SYN, etc) due to contention, the TCP session will not be successful (both the client and the server

do not get any control packets within the initial RTO of 3 seconds) and then the FTP file transfer will not be initiated.

Table 6-6 further shows that the UDP based application performance (delay and jitter) does not improve when using RDM paths with replicating data. This is due to the contention that builds at the end nodes when communicating with the FF-1 and the Command Post. On the other hand, FTP shows lower response time since TCP does not trigger more retransmissions and timeouts when using RDM routes. This is due to a higher probability of loss recovery with replicating data. In summary, when considering the number of packets lost, it shows that RDM with replicating increases the reliability of communications for both UDP and TCP based applications, compared to SP routing.

Table 6-6 Performance of applications & routing protocols– Scenario 3

	SP	RDM	
Sensor Data: End-to-end delay, ms	71.25 ±47.49	73.91 ±33.33	
Audio: End-to-end delay, ms	178.81 ±58.91	199.82 ±40.27	
Audio: jitter, ms	10.79 ±0.65	13.98 ±1.78	
File transfer: DRT, sec	49.18 ±9.28	39.96 ±12.45	
Total attempts of RREQs (generated from both FF-1 & Command Post)	46 ±118	23 ±57	
Average route discovery time, sec	304.51 ±176.78	392.14 ±211.57	
% of utilization of paths when using RDM	-	SP 52%	RDM 48%

6.4 Conclusion

Fire-fighting is a very demanding application area for communication systems. In this chapter, a new approach of replicating packets and transmitting them simultaneously on multiple paths created through an ad hoc networking protocol is presented. The feasibility of this approach is demonstrated by investigations and simulations performed on multipath routing, in particular considering the interference imposed by simultaneous transmission on multiple paths. Results prove the feasibility of using multiple paths for transmitting replicated packets.

Replication causes increased congestion in networks, especially when using applications with higher data rates and when replicating data among paths which are interfering with each other. But with fire-fighting applications, most of the identified applications do not have much data to transmit over multiple hops. Analysis of results show that replicating data over multiple RDM paths enhances the reliability of communications in fire-fighting applications, compared to the use of SP routing. In order to enhance the reliability of the communications, the deployment of more stationary wireless nodes is proposed. This makes it easier to discover RDM paths and hence increase the reliability by replicating data among non-interfering routes.

7. Analytical Model: Determination of RDM Routes

This chapter proposes an analytical model to determine the RDM routing paths in a given wireless multi-hop network by modeling the mutual interference between paths together with Background Traffic Load (BTL). It starts with the review of previous work and highlights what is newly proposed in this work. This is followed by a detailed description of the analytical model including several examples. Then, different types of stationary topologies are used to evaluate the analytical results with the simulation results. This chapter concludes by discussing further enhancements to the proposed solution.

7.1 Related Work

Research in the area of estimating network throughput in wireless multi-hop environments analytically are mainly focused on Single Path (SP) multi-hop ad hoc networks, considering the effect of hidden node and exposed node problems [20, 94-96]. Simultaneous use of multipath, in contrast to SP, may result in further throughput degradation due to the mutual interference between paths. There are a few research papers published that assess network throughput behavior analytically by taking interference into account when using multipath routes. Most of the research uses the max-flow problem [97] as a basis, which considers the flow allocation in wired networks. Reference [98] presents how to find sustainable throughput by using the max-flow problem and extending it to add interference related constraints. These constraints are found using the conflict graph theory. It gives upper and lower bounds for the computation of a sustainable throughput. Reference [99] also uses the max-flow problem with interference constraints. In contrast to [98], interference constraints are computed using Farkas' Lemmas, giving only a tighter upper bound of the throughput. Reference [100] presents a stochastic framework to model the impact of interference between nodes. Instead of the max-flow problem, it uses a non-linear programming model incorporating the interference to compute the network throughput. In contrast to [98] and [99], this paper investigates the variation of sustainable throughput in regular topologies with different types of multipath routes. It concludes that use of multipath with moderate interference performs better than SP, whereas SP performs better than multipath with heavily interfering nodes.

In summary, the above mentioned papers analyze the network throughput for a given set of (fixed) multipath routes. As of this writing, there is no research done to investigate how to model the determination of routing paths (i.e. to select multiple routes) which are completely free of mutual interference or paths that have minimum mutual interference. The work presented here proposes the modeling of such a routing mechanism described as RDM routing. The determination of RDM paths are based on

mutual interference between paths and the BTL of a path. The mutual interference refers to the impact of the interference between links in different paths. BTL determines the number of packets that have already been transmitted in a path. In summary, this work proposes the following new features, compared to previous work:

- Previous work [98-100] computes the sustainable throughput only considering the effect of interference. This thesis extends the computation to consider both the interference and BTL. Furthermore, the model proposed in this thesis considers the effect of interference from other links which are already carrying traffic, but are not part of the selected routes.
- Previous work [98-100] does not provide a solution to discover multiple paths, though it discusses how to determine sustainable throughput when using multipath. This work introduces discovering the best pair of routes which gives the highest sustainable throughput. The throughput is computed considering the effect of interference when using paths simultaneously and also the effect of BTL in a path.
- This thesis analyzes scenarios considering different types of multipath routes (interfering routes: Non RDM, non-interfering routes: Full RDM and partially interfering routes: PRDM routes). In previous work, analysis of behavior of different types of multipath routes is not done except in [100], but it analyzes scenarios without considering BTL.
- In [98], multiple paths are set up with static routes and SP is evaluated using the AODV protocol [12]. None of previous work gives simulation results taken by implementing the respective proposed models to compare against analytical results. This thesis compares analytical results with simulation results considering the effect of interference and BTL when selecting paths. Previous work analyzes network performance by considering only unidirectional traffic. This thesis provides the comparison for both unidirectional and bidirectional traffic using the simulation results, showing a difference in the performance gain.

7.2 Analytical Model: Determination of RDM Routes

This section explains how to determine RDM routes analytically based on two criteria: mutual interference of a path and the BTL of a path. The explanations are given based on a simple 3x3 grid network for the simplicity of understanding. It begins with an overview of the model description providing some background and terminology.

7.2.1 Model Description

Figure 7-1 (a) shows the connectivity graph of a 3x3 grid topology. Each node can communicate with its lateral neighbors, not with diagonal ones. Both the transmission range and the interference range are equal to one unit of distance between two lateral nodes. Nodes n_S and n_D are chosen as the sender and the destination respectively. It is obvious visually that only one possible combination of 2 interference free routes (i.e. FRDM routes) exist for this example network. They are "P_1 {consisting of nodes n_3, n_6 & n_7} & P_3 {consisting of nodes n_1, n_2 & n_5}" which can be used simultaneously

to distribute the traffic among RDM routes. The numbering of paths is done according to the order of paths discovered by the logic explained in section 7.3.

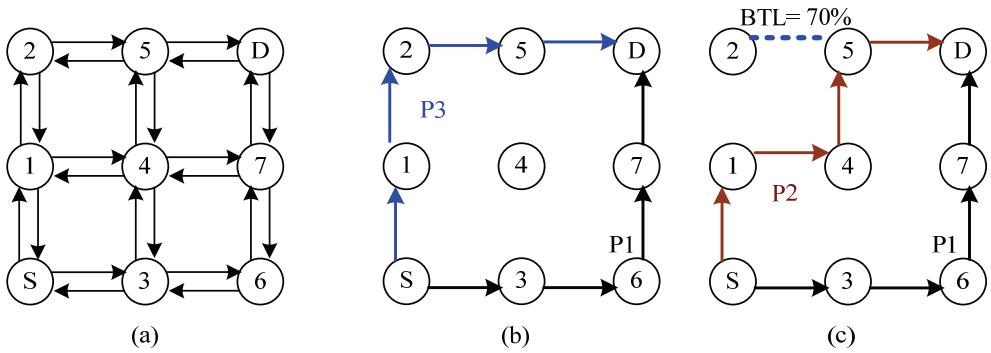


Figure 7-1 3x3 grid (a) all possible links (b) no BTL (c) with 70% of l_{25} is utilized for BTL

As shown in Figure 7-1, case (b) there is no BTL on each link and in case (c) 70% of l_{25} is already being used for background traffic. The selection of paths "P_1 & P_3" should give a higher throughput than selecting P_1 & P_2 if there is no existing traffic (i.e. case (b)) in the network. The selection of P_1 & P_2 should give lower throughput due to the existence of mutual interference. On the other hand, for the case (c), the selection of P_1 & P_3 causes more throughput degradation due to the existing BTL on P_3. In this situation, P_1 & P_2 should perform better though they have interfering links.

The motivation of this work is to discover the best pair of RDM routes by analyzing the interference of a path and the BTL of a path. The best pair of RDM routes is determined by selecting paths giving the highest sustainable throughput. The sustainable throughput is computed considering the interference of a network and the available link capacities. The available link capacity is computed considering the effect of BTL on each link. This model assumes that there exists a central entity which can schedule the packets for transmissions at each node and the data transmission is unidirectional. The discovered RDM routes are used to split the packets of a single flow.

The next sub section gives an overview on terminology used and the analytical model is explained in detail in section 7.2.3. The graph theory and the max-flow problem used in [98] is extended in this work. The graph theory is used to model the interference of a network and the max-flow problem is used to compute the sustainable throughput for a given pair of routes. More details about these theories can be found in [98, 101].

7.2.2 Terminology

This subsection details the theory and terminology used in later sections when discussing the analytical model to determine the RDM paths.

7.2.2.1 Graph Theory

Connectivity Graph: Figure 7-2 shows the connectivity graph of the given wireless chain network. A directional link (e.g. l_{01}) is drawn from one node (e.g. n_0) to another (e.g. n_1), if the communication is possible between n_0 and n_1.

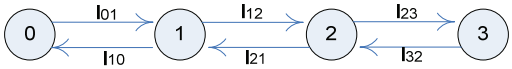


Figure 7-2 Connectivity graph

Conflict Graph: Figure 7-3 shows the corresponding conflict graph [101]. In this graph, the vertices correspond to the links in the connectivity graph, and there is an edge between two vertices, if they cannot be active at the same time, i.e. these two edges are conflicting with each other. For example, link l_{01} is conflicting with l_{21} because the node n_1 could not receive packets from n_0 and n_2 simultaneously. Link l_{01} also conflicts with l_{12} because n_1 cannot send and receive data at the same time. If l_{01} and l_{10} are active at the same time, then each node of n_0 and n_1 should be active as a sender and a receiver at the same time, and this could never happen. When n_0 is transmitting to n_1, n_2 cannot starts transmitting since it interfere with the receipt of packets at n_1. Therefore l_{01} and l_{23} cannot active at the same time.

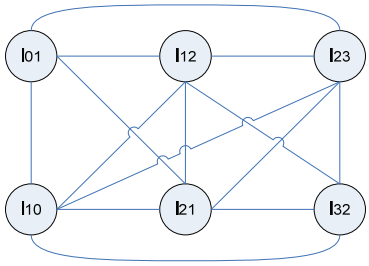


Figure 7-3 Conflict graph

Generating a Conflict Graph based on the Physical Interference Model: In the physical interference model, the judgment whether a transmission from one node to another is successful is based on the Signal to Noise Ratio (SNR). Suppose node n_i wants to transmit a packet to node n_j. The total noise N_j at node n_j consists of ambient noise N_a and the interference from other ongoing transmissions inside the network. The transmission is considered as successful, if $SNR_{ij} \geq SNR_{thresh}$, where

SNR_{ij} denotes the SNR at the node n_j, and SNR_{thresh} is the threshold of SNR. According to the physical interference model, the maximum permissible noise can be represented as in (7-1), where SS_{ij} denotes the signal strength at the node n_j due to the transmissions from node n_i. This consists of ambient noise and the maximum permissible noise imposed by the interference from other communications in the network.

$$\frac{SS_{ij}}{SNR_{thresh}} \qquad \text{7-1}$$

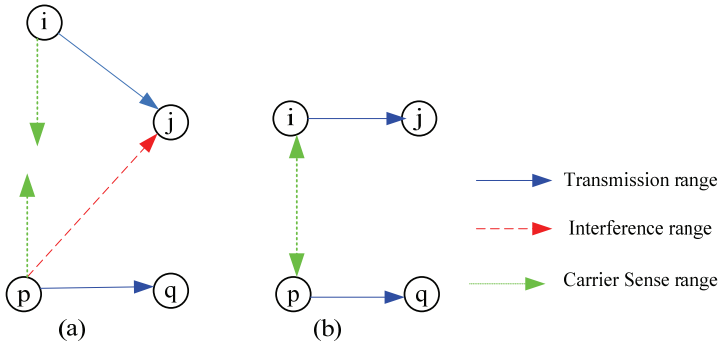


Figure 7-4 (a) two transmitters are outside CS range (b) two transmitters are inside CS range

Figure 7-4 (a) shows two links l_{ij} and l_{pq}, where node n_i is transmitting to n_j, and simultaneously node n_p is sending packets to n_q. In this case, the link l_{ij} is taken as the target link, and link l_{pq} interferes with link l_{ij}. The data transmission over link l_{ij} is possible, even though node n_j is suffering from the strong interference imposed by node n_p. Eq. (7-2) shows the weight factor ω_{ij}^{pq} that can be computed based on the physical interference model using (7-1). Weight factor, ω_{ij}^{pq} indicates how strong the interference between the link l_{pq} and link l_{ij} is.

$$\omega_{ij}^{pq} = \frac{SS_{pj}}{\frac{SS_{ij}}{SNR_{thresh}} - N_a} \qquad \text{7-2}$$

SS_{pj} and SS_{ij} denote the signal strength at the node n_j due to the transmissions from node n_p and node n_i respectively. SNR_{thresh} is the SNR threshold and N_a is the ambient noise. The part of $\frac{SS_{ij}}{SNR_{thresh}} - N_a$ denotes the *maximum permissible interference noise* at

node n_j that still allows a successful reception of node n_i's transmissions. Assuming the ambient noise N_a is negligible; (7-2) is re-written as (7-3). The SNR threshold SNR_{thresh} is dependent on the hardware and the PHY mode used. The signal strength measured at the receiver, node n_j, i.e. SS_{pj} and SS_{ij}, is determined by 3 factors: the *transmitting power* P_{Tx} at the sender, the *path loss* P_{Loss} during the transmission, and the *receiver sensitivity* Sen_{Rx}.

$$\omega_{ij}^{pq} = \frac{SS_{pj}}{\frac{SS_{ij}}{SNR_{thresh}}} \qquad 7\text{-}3$$

The path loss P_{Loss} *is* computed by using a free space propagation model, assuming no obstacle between the sender and the receiver. (7-4) is used for calculating the path loss [102], in which f stands for the frequency in GHz and d stands for the distance between the transmitter and receiver in km.

$$P_{Loss} = 92.45 + 20\log_{10} f + 20\log_{10} d \qquad 7\text{-}4$$

If P_{Tx} at node n_i, P_{Loss} between node n_i and n_j, and Sen_{Rx} at node n_j are known, the signal strength, SS_{ij} can be computed as in (7-5).

$$SS_{ij} = (P_{Tx})_i - (P_{Loss})_{ij} - (Sen_{Rx})_j \qquad 7\text{-}5$$

In Figure 7-4 (b) the distance between node n_i and n_p, is within the carrier sensing range in contrast to Figure 7-4 (a). Both node n_i and node n_p are sharing the same channel in a multi-hop ad hoc network. According to CSMA/CA in IEEE 802.11 technologies [103], the node n_i does not detect the medium as free while node n_p is transmitting data to node n_q. In other words, if the node n_i wishes to send data to node n_j, it has to sense the channel first. If the channel is sensed "idle", node n_i is permitted to transmit data; otherwise, if the medium is sensed as "busy", node n_i has to backoff its transmission. In this situation, transmissions over links l_{ij} and l_{pq} cannot be active simultaneously, when using CSMA/CA wireless technologies. Therefore, the weight factor ω_{ij}^{pq} is also not computed if both transmitters are within the carrier sensing range as shown in Figure 7-4 (b).

7.2.2.2 Computation of Independent Sets

With the knowledge about the connectivity graph and the conflict graph, this section introduces another concept used in graph theory called Independent Sets (IS). An IS,

computed in this thesis, denotes a group of links, which can be active simultaneously without interfering with each other.

Given a graph, e.g. the conflict graph in Figure 7-3, an IS is defined as a set V of vertices such that for every two vertices in V, there is no edge connecting them [101]. The size of an IS is defined as the number of vertices it contains. A *maximum independent set* is the largest independent set in the given graph, and adding any other vertex introduces edges between vertices in it. For example, the link l_{01} does not conflict with l_{32}, so that these two links can be placed into one IS as $IS_1 = \{l_{01}, l_{32}\}$ for Figure 7-3. The ISs are computed in this thesis using the weight factors that are computed based on the physical interference model (see section 7.2.3).

It is not always possible to find all ISs of a given graph. When the graph's size increases, the process of finding all its ISs requires more computational resources. A difficulty is to find the *maximum independent sets*, which have the maximum size. Finding of *maximum independent sets* may reduce the total number of ISs drastically. Finding all ISs of a given graph is *NP-hard*. The computation of all ISs in a given network is detailed in Figure 9-2 of Appendix I.

7.2.2.3 Extended Max-Flow Problem

Equation 7-6 shows the standard max-flow problem [97] with additional 2 constraints to consider the wireless interference. The max-flow problem is used to describe the method of finding the maximum throughput of a given wired network. Assuming the given network is a closed network, i.e. there are no packet losses, and then the problem of finding the maximum network throughput can be mapped to maximising the outgoing data flows from the source to all its neighbouring nodes.

The first 5 constraints in equation 7-6 represent the standard max-flow problem. f_{ij} denotes the amount of flow on link l_{ij}, Cap_{ij} denotes the link capacity of link l_{ij}, and L_C is the set of all links in the connectivity graph C. The maximisation shows that the sum of flows out of the source is to be maximized. The constraint <1> states the flow conservation, i.e. at each node, except the source and the destination; the amount of the incoming flow is equal to the outgoing flow. The constraint <2> shows that there is no flow going into the source. The constraint <3> says that accordingly there is no flow coming out of the destination. The constraint <4> indicates that the amount of flow on a link is limited by the link capacity. The constraint <5> shows that the value of a flow must be non-negative. This is extended by adding the last two constraints related to the interference between links, using the ISs. In this model, scheduling of ISs is used to avoid the interference between conflicting links, when computing the sustainable throughput. Given a network, and there exist K ISs, e.g. $IS_1, IS_2, \ldots IS_K$. λ_n denotes the fraction of time that can be used by the independent set IS_n. This represents that

each IS can only be active during a part of the total transmission time, which is considered as 1 unit. The constraint <6> means that the sum of the active time of all ISs is limited by 1; and the constraint <7> shows that the flow f_{ij} over the link l_{ij} is related to the ISs, to which it belongs. Note that one link can be included in more than one IS. All links belonging to independent set IS_n can be simultaneously active for λ_n faction of time, and it has been required that the sum of all λ_n is limited by 1. The constraint <4> of the standard max-flow problem is replaced by the constraint <7>. If a link is already carrying the traffic, Cap_{ij} of the link l_{ij} in the constraint <7> has to be modified to reflect the available capacity of a link by deducting the amount of traffic used for BTL.

$$
\begin{aligned}
&\max \sum_{l_{si} \in L_C} f_{si} \\
&\textit{subject to} \\
&\sum_{l_{ij} \in L_C} f_{ij} = \sum_{l_{ji} \in L_C} f_{ji} \quad \forall n_i \in N_C \setminus \{n_s, n_d\} \,\&\, \forall n_j \in N_C \quad < 1 > \\
&\sum_{l_{is} \in L_C} f_{is} = 0 \quad < 2 > \\
&\sum_{l_{di} \in L_C} f_{di} = 0 \quad < 3 > \\
&f_{ij} \le Cap_{ij} \quad \forall l_{ij} \in L_C \quad < 4 > \\
&f_{ij} \ge 0 \quad \forall l_{ij} \in L_C \quad < 5 > \\
&\sum_{n=1}^{K} \lambda_n \le 1 \quad < 6 > \\
&f_{ij} \le \sum_{l_{ij} \in IS_n} \lambda_n Cap_{ij} \quad < 7 >
\end{aligned}
\qquad \text{7-6}
$$

7.2.3 Interference Computation: 3x3 Grid Topology

As the first step, all possible paths from the sender to the destination should be found together with the interference of each link in a path. Here, the interference between links is computed using a conflict graph. The weighted conflict graph is generated, assuming all the nodes are transmitting simultaneously. The weight factors are computed using a physical interference model as explained in section 7.2.2.1. As shown in Figure 7-1 (a), there exist 24 links due to the size of the corresponding conflict graph. The equivalent conflict matrix is given in Table 7-1 and Table 7-2. The computation of weight factors for the 3x3 grid network should be considered for mainly 3 different types of links as shown below.

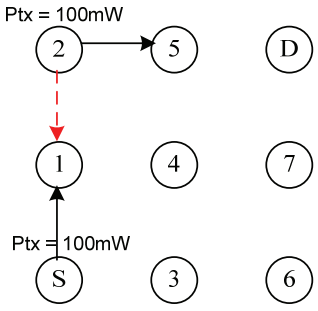
7.2.3.1 Simultaneously Not Active Links

If there are two links that cannot be active simultaneously, the weight factor is not computed. This is indicated as "-" in the conflict matrix. There are two possibilities for having such links.

- If 2 links have one common node, then they cannot be active at the same time. E.g. the link l_{S3} and the link l_{S1} (node n_S cannot transmit simultaneously to both node n_1 and node n_3) or the link l_{1S} and link l_{3S} (node n_S cannot receive simultaneously from both node n_1 and node n_3). The latter case is similar to reality when avoiding the hidden node problem in WLAN with RTS/CTS messages.
- If senders of both links are within each other's carrier sensing range (as shown in Figure 7-4 (b)), they cannot also be active simultaneously due to the CSMA/CA mechanism used in the MAC protocol. E.g. both node n_s and node n_1 are in each other's CS range, so the link l_{S3} and the link l_{12} cannot be active simultaneously.

7.2.3.2 Simultaneously Active Interfering Links

There are links that can be active simultaneously, but they can be interfering with each other's transmission. For example, one link's destination is within another sender's interference range. In this situation, the weight factor has to be computed using (7-3). Figure 7-5 gives an example of such links for the computation of weight factor, ω_{S1}^{25}.



$$\omega_{S1}^{25} = \frac{SS_{21}}{\frac{SS_{S1}}{SNR_{thresh}}} = \frac{0.21dBm}{\frac{0.21dBm}{4dB}} = 2.51$$

Figure 7-5 Computation of ω_{S1}^{25} – the target link l_{s1} and interfering link is l_{25} (ω_{S1}^{25} indicates how strong the interference from l_{25} to l_{s1})

In order to compute the weight factors, values used in simulated scenarios have been used. Simulations are done using PHY mode set to 1 Mbps. Other parameters used are Sen_{Rx} is set to *-76dBm*, P_{Tx} is set to *100 mW* and f is set to *2.45 GHz* and SNR_{thresh} is set to *4dB*. P_{Loss} at n_1 due to the communication between n_2 and n_5 can be computed

using (7-4). The distance between n_2 and n_1 is set to 1 unit equal to 0.6 km in the simulator. All nodes in the network have identical properties.

$$P_{Loss} = 92.45 + 20\log_{10} 2.45 + 20\log_{10}(0.6) = 95.79dB$$

(7-5) is used to compute the received signal strength at n_1, SS_{21} and SS_{S1} due to the transmission from n_2 and n_s,

$$SS_{21} = 20dBm - 95.79dB - (-76dBm) = 0.21dBm = 1mW$$

Similarly, the distance between n_S and n_1 is also 1 unit, both SS_{21} and SS_{S1} have a similar value. As shown in Figure 7-5, the weight factor ω_{s1}^{25}, between the link l_{S1} and the link l_{25} can be computed as 2.51.

7.2.3.3 Simultaneously Active Non-Interfering Links

If both the sender and the destination of the link under study are out of the interference range, these links do not interfere with each other. For example, node n_1 cannot interfere with n_7's transmission. P_{Loss} at n_1 from the communication between n_7 and n_D can be computed using (7-4). The distance between n_1 and n_7 is 2 units (i.e. 1.2 km).

$$P_{Loss} = 92.45 + 20\log_{10} 2.45 + 20\log_{10}(1.2) = 101.817dB$$

The received signal strength at n_1 due to the transmission from node n_7 is computed as:

$$SS_{71} = 20dBm - 101.817dB - (-76dBm) = -5.817dBm = 0.262mW$$

As computed earlier, $SS_{S1} = 1mW$. Therefore the interference to the link, l_{S1} from link l_{7D} can be computed using (7-3):

$$\omega_{S1}^{7D} = \frac{SS_{71}}{\frac{SS_{S1}}{SNR_{thresh}}} = \frac{0.262}{\frac{1.05}{2.51}} = 0.63$$

Similarly, all the weight factors can be computed for the connectivity graph of Figure 7-1 (a). This can be represented in a matrix as shown in Table 7-1 and Table 7-2. In this conflict matrix, columns represent target links while rows represent interfering links. In order to find interfering links, the weight factors have to be normalized w.r.t. SNR_{thresh} (i.e. 2.51 for this example). After normalizing, weight factors of 2.51 refers to 1 and all the other weight factors computed in this example show the value less than 1. Therefore, weight factors of 2.51 refer to interfering links, while all the other weight factors refer to non-interfering links.

Table 7-1 Conflict matrix of a 3x3 grid topology – Part 1 (the shaded weight factors represent the interfering links)

link	S-1	S-3	1-S	1-2	1-4	2-1	2-5	3-S	3-4	3-6	4-1	4-3
S-1	-	-	-	-	-	-	0.50	-	-	-	-	2.51
S-3	-	-	-	-	-	2.51	0.50	-	-	-	2.51	-
1-S	-	-	-	-	-	-	-	-	2.51	0.50	-	-
1-2	-	-	-	-	-	-	-	2.51	2.51	0.50	-	-
1-4	-	-	-	-	-	-	-	2.51	-	0.50	-	-
2-1	-	0.50	-	-	-	-	-	0.63	1.25	0.31	-	0.50
2-5	2.51	0.50	-	-	-	-	-	0.63	0.25	0.31	2.51	0.50
3-S	-	-	-	0.50	2.51	1.25	0.63	-	-	-	-	-
3-4	-	-	2.51	0.50	-	1.25	0.63	-	-	-	-	-
3-6	-	-	2.51	0.50	2.51	1.25	0.63	-	-	-	-	-
4-1	-	2.51	-	-	-	-	2.51	-	-	-	-	-
4-3	2.51	-	-	-	-	2.51	2.51	-	-	-	-	-
4-5	2.51	2.51	-	-	-	2.51	-	-	-	-	-	-
4-7	2.51	2.51	-	-	-	2.51	2.51	-	-	-	-	-
5-2	1.25	0.63	0.50	-	2.51	-	-	0.50	2.51	0.50	-	-
5-4	1.25	0.63	0.50	2.51	-	-	-	0.50	-	0.50	-	-
5-D	1.25	0.63	0.50	2.51	2.51	-	-	0.50	2.51	0.50	-	-
6-3	0.63	-	0.63	0.31	1.25	0.50	0.50	-	-	-	0.50	-
6-7	0.63	2.51	0.63	0.31	1.25	0.50	0.50	-	-	-	0.50	2.51
7-4	0.63	1.25	0.50	0.50	-	0.63	1.25	0.50	-	2.51	-	-
7-6	0.63	1.25	0.50	0.50	2.51	0.63	1.25	0.50	2.51	-	-	-
7-D	0.63	1.25	0.50	0.50	2.51	0.63	1.25	0.50	2.51	2.51	-	-
D-5	0.50	0.50	0.31	0.63	1.25	0.50	-	0.31	1.25	0.63	0.50	0.50
D-7	0.50	0.50	0.31	0.63	1.25	0.50	2.51	0.31	1.25	0.63	0.50	0.50
No: of Interfering links	4	4	2	2	6	4	4	2	6	2	2	2

Table 7-2 Conflict matrix of a 3x3 grid topology – Part 2 (the shaded weight factors represent the interfering links)

link	4-5	4-7	5-2	5-4	5-D	6-3	6-7	7-4	7-6	7-D	D-5	D-7
S-1	0.50	0.50	0.63	1.25	0.31	2.51	0.50	1.25	0.63	0.31	0.50	0.50
S-3	0.50	0.50	0.63	1.25	0.31	-	0.50	1.25	0.63	0.31	0.50	0.50
1-S	-	-	2.51	2.51	0.50	1.25	0.63	2.51	0.50	0.50	1.25	0.63
1-2	-	-	-	2.51	0.50	1.25	0.63	2.51	0.50	0.50	1.25	0.63
1-4	-	-	2.51	-	0.50	1.25	0.63	-	0.50	0.50	1.25	0.63
2-1	2.51	0.50	-	-	-	0.50	0.63	1.25	0.31	0.63	2.51	0.50
2-5	-	0.50	-	-	-	0.50	0.63	1.25	0.31	0.63	-	0.50
3-S	-	-	0.50	2.51	0.50	-	-	2.51	2.51	0.50	0.63	1.25
3-4	-	-	0.50	-	0.50	-	-	-	2.51	0.50	0.63	1.25
3-6	-	-	0.50	2.51	0.50	-	-	2.51	-	0.50	0.63	1.25
4-1	-	-	-	-	-	2.51	2.51	-	-	-	2.51	2.51
4-3	-	-	-	-	-	-	2.51	-	-	-	2.51	2.51
4-5	-	-	-	-	-	2.51	2.51	-	-	-	-	2.51
4-7	-	-	-	-	-	2.51	-	-	-	-	2.51	-
5-2	-	-	-	-	-	0.63	0.63	2.51	0.50	2.51	-	-
5-4	-	-	-	-	-	0.63	0.63	-	0.50	2.51	-	-
5-D	-	-	-	-	-	0.63	0.63	2.51	0.50	-	-	-
6-3	0.63	2.51	0.31	1.25	0.63	-	-	-	-	-	0.50	2.51
6-7	0.63	-	0.31	1.25	0.63	-	-	-	-	-	0.50	-
7-4	-	-	0.50	-	2.51	-	-	-	-	-	-	-
7-6	-	-	0.50	2.51	2.51	-	-	-	-	-	-	-
7-D	-	-	0.50	2.51	-	-	-	-	-	-	-	-
D-5	-	2.51	-	-	-	0.50	2.51	-	-	-	-	-
D-7	2.51	-	0.63	-	-	0.50	-	-	-	-	-	-
No: of Interfering links	2	2	2	6	2	4	4	6	2	2	4	4

The conflict matrix is also used to compute the ISs between 2 given pairs of routing paths. It also shows how many links are interfering with each single link in the network. For this example, links toward the middle of the network (l_{14}, l_{34}, l_{54} and l_{74}) are interfering more with other links.

7.2.4 Selection of a Pair of RDM Routing Paths

There are 12 possible combinations of paths existing for the connectivity between n_S and n_D as shown in Table 7-3. Table 7-3 shows computed values for all paths of Figure 7-1 case (a). It can be observed that since the node n_4 is located in the middle of the grid network, the links connected to this node are interfering more.

$$T_r = \max(T_{1r}, T_{2r}, ..T_{ir}, T_{k,r}) \qquad 7\text{-}7$$

The BTL of a path can be computed as in (7-7). T_r denotes the background traffic of the r^{th} path given as a percentage w.r.t the available link capacity. T_{ir} is equal to the traffic load of the i^{th} link in the r^{th} path. k_r represents the number of links in the r^{th} path. The BTL of the r^{th} path, T_r is considered as the maximum BTL of an individual link. For example, BTL of P_3 in Figure 7-1 - case (c) is equal to the highest BTL of 0.7 in link l_{25}.

Table 7-3 Possible paths of a 3x3 grid topology of Figure 7-1 case (a)

Path id	Used Links	# of Interfering Links	BTL	Hop Count
P1	S-3, 3-6, 6-7, 7-D	12	0	4
P2	S-1, 1-4, 4-5, 5-D	14	0	4
P3	S-1, 1-2, 2-5, 5-D	12	0	4
P4	S-1, 1-2, 2-5, 5-4, 4-7, 7-D	20	0	6
P5	S-1, 1-2, 2-5, 5-4, 4-3, 3-6, 6-7, 7-D	26	0	8
P6	S-1, 1-4, 4-7, 7-D	14	0	4
P7	S-1, 1-4, 4-3, 3-6, 6-7, 7-D	20	0	6
P8	S-3, 3-4, 4-1, 1-2, 2-5, 5-D	20	0	6
P9	S-3, 3-4, 4-5, 5-D	14	0	4
P10	S-3, 3-4, 4-7, 7-D	14	0	4
P11	S-3, 3-6, 6-7, 7-4, 4-1, 1-2, 2-5, 5-D	26	0	8
P12	S-3, 3-6, 6-7, 7-4, 4-5, 5-D	20	0	6

7.2.4.1 Computation of Sustainable Throughput

Once all the possible routing paths are known, the pair of routes to be used as RDM paths is chosen by finding the pair which gives the highest sustainable throughput. This requires the computation of the ISs for each pair of discovered routing paths. The previously computed conflict matrix can be used to find ISs of each pair. ISs are computed using the algorithm given in section 7.2.2.2. After knowing the ISs, all constraints in the extended max-flow problem as given in (7-6) can be completed. The

details of the computation of sustainable throughput is given for the pair of P_1 and P_3 and Table 7-4 shows all the ISs that are found for the use of P_1 and P_3 simultaneously.

Figure 7-6 shows the input variables of all constraints given to the linear program for the 3x3 grid topology shown in Figure 7-1 case (b) , where f_{ij} represents the flow capacity between node n_i and node n_j. The discovered 10 ISs are represented as IS_1, IS_2, .., IS_{10} and λ_n represents the fraction of active time allocated for IS_n. The link capacity is defined as 1 unit. Here the objective is to maximize the flow originated from node n_S, that is split among P_1 and P_3. Constraint 1 shows that there are no packet losses. Constraints 2 and 3 show that the data transmission from the sender, n_S to the destination, n_D is unidirectional. Constraint 5 shows that each flow capacity must be non negative. Constraint 6 shows that the sum of active time of all ISs has to be limited by 1. Constraint 7 show that amount of flow in a link is limited by the active time of ISs that a corresponding link consists of. E.g., link l_{S1} is included in both IS_1 and IS_2. Therefore, flow capacity f_{S1} of l_{S1}, has to be limited by the active time allocated for IS_1 and IS_2 (i.e. λ_1 and λ_2).

Table 7-4 Total ISs found for a pair of P_1 and P_3 in 3x3 grid topology – shaded ISs are used for optimal scheduling

IS id	Links
IS_1	l_{S1}, l_{5D}, l_{67}
IS_2	l_{S1}, l_{7D}
IS_3	l_{12}, l_{36}
IS_4	l_{12}, l_{67}
IS_5	l_{12}, l_{7D}
IS_6	l_{S3}, l_{25}, l_{7D}
IS_7	l_{25}, l_{36}
IS_8	l_{25}, l_{67}
IS_9	l_{S3}, l_{5D}
IS_{10}	l_{36}, l_{5D}

Table 7-5 shows the sustainable throughput of subset of different pair of paths existing in a 3x3 grid topology. The sustainable throughput of P_1 and P_3 is computed as 0.667, i.e. 66.7% of the link capacity. This is achieved by scheduling the links in 3 ISs (IS_1, IS_3 & IS_6) out of total 10 ISs discovered as given in Table 7-4. IS_1 and IS_6 are maximum ISs, containing the maximum number of links that can be active

simultaneously. The traffic is evenly distributed between the RDM paths of P_1 and P_3. The ISs in selected RDM paths are assigned an equal active time of 1/3 when scheduling ($\lambda_1 = \lambda_3 = \lambda_6 = 0.333$).

How packets are transmitted with scheduling is shown in Figure 7-7-(a). The use of P_1 and P_2 gives different scheduling as shown in Figure 7-7-(b). The sustainable throughput in this case is computed as 0.4285. It uses 5 ISs as shown in Table 7-5. When the paths are not fully RDM, less links are included in the IS, which lead to lower sustainable throughput.

```
fS_3 + fS_1;                                                    } Objective function
/*------------------------*/
fS_1 = f1_2;
fS_1 = f2_5;
fS_1 = f5_D;
fS_3 = f3_6;                                                    } Constraint 1
fS_3 = f6_7;
fS_3 = f7_D;
/*------------------------*/
f3_S + f1_S = 0;                                                } Constraint 2
/*------------------------*/
fD_5 + fD_7 = 0;                                                } Constraint 3
/*------------------------*/
fS_1 > 0;
f1_2 > 0;
f2_5 > 0;
f5_D > 0;
fS_3 > 0;                                                       } Constraint 5
f3_6 > 0;
f6_7 > 0;
f7_D > 0;
/*------------------------*/
λ1 + λ2 + λ3 + λ4 + λ5 + λ6 + λ7 + λ8 + λ9 + λ10 < 1;          } Constraint 6
/*------------------------*/
fS_1 < λ1 + λ2 ;
f1_2 < λ3 + λ4 + λ5;
f2_5 <λ6 + λ7 + λ8;
f5_D < λ1 + λ9 + λ10;                                           } Constraint 7
fS_3 < λ6 + λ9;
f3_6 < λ3 + λ7 + λ10;
f6_7 < λ1 + λ4 + λ8;
f7_D < λ2 + λ5 + λ6;
/*------------------------*/
```

Figure 7-6 Constraints in the linear program model to find sustainable throughput for 3x3 grid topology of Figure 7-1 case (b)

The highest sustainable throughput is given by the pair of P_1 & P_3, which is node disjoint as well as FRDM routing paths having no common interfering links. The next

highest throughput of 42.85% is given when using P_1 & P_2, which are node disjoint, but having mutual interference creating PRDM routing.

The paths which have only one sharing link provides the next highest throughput (e.g. P_1 & P_4 and P_1 & P_6). These paths are also PRDM, but with a common shared link. When two paths have more than one common link (e.g. P_1 & P_7) or the first link with the originating node (e.g. P_1 & P_8) is shared, the sustainable throughput is obtained only using a SP. These results show that the use of non-interfering node disjoint routes (i.e. FRDM routes) provide the maximum sustainable throughput, while the sustainable throughput of PRDM routes becomes lower when more interfering links are present. Furthermore, this shows that SP is better than using multiple paths simultaneously if paths have more shared and interfering links.

Table 7-5 Sustainable throughput of sub set of different pair of paths in Figure 7-1 case (a)

Path id	Used Links	Sustainable Throughput (%)	ISs used for optimal scheduling with relevant time fraction
P_1 & P_2	S-3, 3-6, 6-7, 7-D & S-1, 1-4, 4-5, 5-D	42.85	$0.2857\{l_{S3}, l_{7D}\}, 0.2857\{l_{36}, l_{5D}\}, 0.1428\{l_{S1}, l_{5D}, l_{67}\}, 0.1428\{l_{14}, l_{67}\}, 0.1428\{l_{45}\}$
P_1 & P_3	S-3, 3-6, 6-7, 7-D & S-1, 1-2, 2-5, 5-D	66.70	$0.3333\{l_{S1}, l_{5D}, l_{67}\}, 0.3333\{l_{12}, l_{36}\}, 0.3333\{l_{S3}, l_{25}, l_{7D}\}$
P_1 & P_4	S-3, 3-6, 6-7, 7-D & S-1, 1-2, 2-5, 5-4, 4-7, 7-D	40	$0.2\{l_{S1}, l_{54}, l_{67}\}, 0.2\{l_{12}, l_{36}\}, 0.2\{l_{S3}, l_{25}, l_{7D}\}, 0.2\{l_{47}\}$
P_1 & P_5	S-3, 3-6, 6-7, 7-D & S-1, 1-2, 2-5, 5-4, 4-3, 3-6, 6-7, 7-D	33.33	$0.3333\{l_{36}\}, 0.3333\{l_{S3}, l_{7D}\}, 0.3333\{l_{67}\}$
P_1 & P_6	S-3, 3-6, 6-7, 7-D & S-1, 1-4, 4-7, 7-D	40	$0.20\{l_{S1}, l_{7D}\}, 0.20\{l_{14}, l_{67}\}, 0.20\{l_{S3}, l_{7D}\}, 0.20\{l_{47}\}, 0.20\{l_{36}\}$
P_1 & P_7	S-3, 3-6, 6-7, 7-D & S-1, 1-4, 4-3, 3-6, 6-7, 7-D	33.33	$0.3333\{l_{S3}, l_{7D}\}, 0.3333\{l_{67}\}, 0.3333\{l_{36}\}$
P_1 & P_8	S-3, 3-6, 6-7, 7-D & S-3, 3-4, 4-1, 1-2, 2-5, 5-D	33.33	$0.3333\{l_{S3}, l_{7D}\}, 0.3333\{l_{34}\}, 0.3333\{l_{47}\}$
P_1 & P_9	S-3, 3-6, 6-7, 7-D & S-3, 3-4, 4-5, 5-D	33.33	$0.3333\{l_{S3}, l_{7D}\}, 0.1111\{l_{36}, l_{5D}\}, 0.1111\{l_{67}, l_{5D}\}, 0.2222\{l_{34}\}, 0.2222\{l_{45}\}$
P_1 & P_{10}	S-3, 3-6, 6-7, 7-D & S-3, 3-4, 4-7, 7-D	33.33	$0.3333\{l_{S3}, l_{7D}\}, 0.3333\{l_{34}\}, 0.3333\{l_{47}\}$
P_1 & P_{11}	S-3, 3-6, 6-7, 7-D & S-3, 3-6, 6-7, 7-4, 4-1, 1-2, 2-5, 5-D	33.33	$0.3333\{l_{S3}, l_{7D}\}, 0.3333\{l_{67}\}, 0.3333\{l_{36}\}$
P_1 & P_{12}	S-3, 3-6, 6-7, 7-D & S-3, 3-6, 6-7, 7-4, 4-5, 5-D	33.33	$0.3333\{l_{S3}, l_{7D}\}, 0.3333\{l_{67}\}, 0.3333\{l_{36}\}$

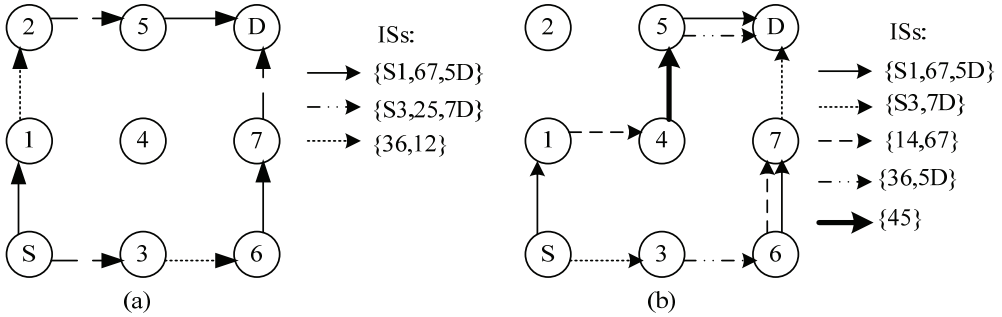


Figure 7-7 Scheduling used (a) P_1 and P_3 (b) P_1 and P_2

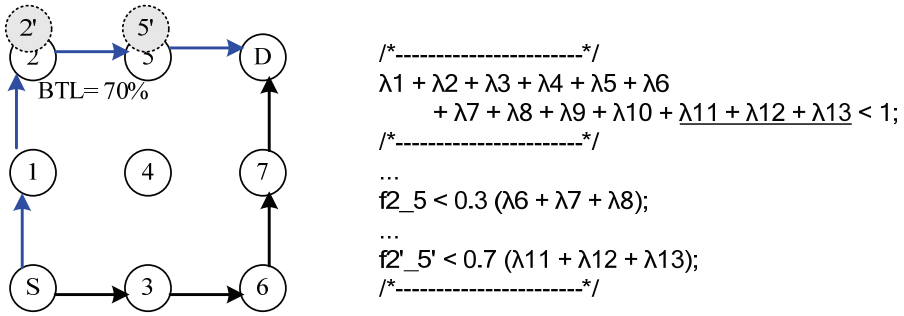


Figure 7-8 Modeling of the effect of BTL between node 2 & node 5 of 3x3 grid topology

7.2.4.2 Computation of Sustainable Throughput when BTL is Present

Above examples show the use of routing paths without any BTL in a link. The BTL affects the throughput of other paths if the links that carry BTL are also shared links or lie within the interfering range. Therefore, the effect of BTL is considered as follows:

- Case 1: The links that carry BTL are sharing links of the active paths considered. For example, P_1 and P_3 used in case (c) in Figure 7-1, the 70% of l_{25} is utilized for BTL. When computing the ISs, the scheduling used for BTL also has to be taken into account. This is represented in the analytical model, by adding another virtual link between n_2 and n_5 during the computation of ISs. Figure 7-8Figure 7-8 shows the virtual link that is used to represent the BTL between n_2 and n_5, represented as $n_{2'}$ and $n_{5'}$. When considering this additional link, 3 more ISs are used to include the scheduling for the BTL. When computing the sustainable throughput, constraint 7 of f_{25}, has to be rewritten as $f_{25} < 0.3\ (\lambda_6 + \lambda_7 + \lambda_8)$ since the available capacity of the link l_{25} is 30%. The new constraint of $f_{2'5'} < 0.7\ (\lambda_{11} + \lambda_{12} + \lambda_{13})$ has to be added to represent the BTL (see Figure 7-8Figure 7-8).
- Case 2: The links that carry BTL are not sharing with the links on the active paths considered. For example, P_1 and P_2 used in case (c) in Figure 7-1, l_{25}

which carries BTL is not an active link for both paths of P_1 and P_2. When computing the ISs, the link l_{25} should also be included together with all the other links of the active paths. When computing the sustainable throughput, constraint 5 and constraint 7 have to be modified to include link l_{25} as well.

Figure 7-9 shows the variation of sustainable throughput with the increase of the percentage of BTL used in the link l_{25} for 3 different types of path combinations; FRDM routes (P_1 and P_3), PRDM with node disjoint routes (P_1 and P_2) and PRDM with shared links (P_1 and P_4). These results show that the performance gain with FRDM routes is degraded with the increase of the BTL on the paths. The performance of PRDM does not depend much on the BTL since it does not share the link l_{25}.

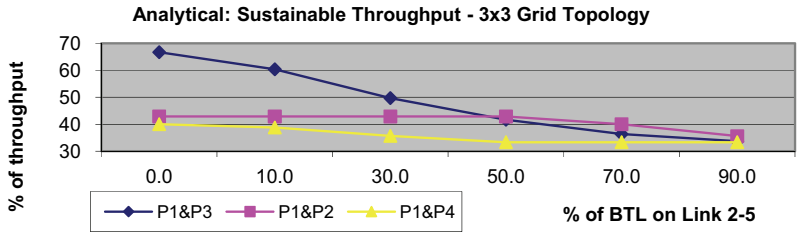


Figure 7-9 Analytical sustainable throughput by varying BTL on l_{25} - 3x3 Grid

7.3 Implementation of Analytical Model

All the computations explained above (computation of the conflict matrix, independent sets, all possible paths and the selection of RDM paths) are implemented using the C++ programming language. The flow chart of this implementation is given in the Appendix I. Two basic classes of *NetworkNode* and *NetworkLink* are used to keep the geometric coordinates of each single node in a given network and to show the relationship between nodes respectively. Discovery of all possible paths between the source and the destination in a given network is done using the following algorithm, avoiding loops in a path.

- Step 1: Starting from the originating node, discover a path to reach the destination node.
  - A. Check possible neighbors of the current node in the following sequence: *Up, Left, Down, Right*.
  - B. If a neighbor is found and this neighbor does not exist in the current path, save this neighbor and take it as the current node.
  - C. Repeat step 1.A)
- Step 2: Once the destination is found as the neighbor, save the path.

- Step 3: Select the last node of the recently discovered path.
- Step 4: If the last node is not equal to the originating node, go to step 1.A.
- Step 5: If the last node is equal to the originating node, discover a path via an immediate hop which has not been considered before.

7.4 Evaluation of Analytical Results

Chapter 3 explains how a pair of RDM routes is discovered in the simulator. Table 7-6 gives an overview to the comparison of different algorithms used in both analytical and simulation environments.

Table 7-6 Algorithms used for simulation and analytical environments

Algorithms	Simulation	Analytical
Num: of paths considered	Selected paths are considered by limiting the RREQ propagation by each intermediate node in order to reduce control traffic.	All possible paths without creating loops are considered.
Selection criteria	A routing pair which gives the lowest Path Load (PL). PL consists of the mutual interference between two paths and the existing BTL in the paths. Since the throughput cannot be measured, until data traffic is initiated in real networks, the best pair of paths is computed based on heuristics. First select the primary path, which has the lowest number of interfering nodes. Then select the secondary path based on the mutual interference w.r.t. the primary path and existing BTL in the path.	A routing pair which gives the highest throughput. Throughput is computed by avoiding the use of interfering links simultaneously. The effect of BTL is also considered when computing the throughput for a given pair of routes, while scheduling the non-interfering links to achieve optimal throughput.
Computation of the interference	The interfering nodes are computed individually. This is done if the interfering signal power is greater than or equal to the receiver sensitivity of a particular node. In this method, aggregated interference from simultaneously active multiple nodes is not computed.	The interference between a pair of links is weighted according to the physical interference model. This computes the effect of aggregated interference if multiple links are active simultaneously.
Computation of the BTL	BTL is computed by monitoring the packets transmitted and received by a node. The weighted average is computed for the total number of packets transmitted and received over a period.	By considering the links that carry BTL in the conflict graph and considering the flows used by the BTL in the max-flow problem

The analytical results are validated with the simulation results by computing the percentage of the sustainable throughput of different topologies. This is measured in the simulated scenarios by computing the maximum number of bits that an application can send until there is packet loss at the link layer. For example, if an application can send a maximum of 20 packets of 1000 bytes each in a second, without losing any packet at the

link layer, the sustainable throughput is computed as "(1000x20)x8 bps". Since the PHY mode is set to 1 Mbps, the percentage of the sustainable throughput is computed as 16%. For analytical results, this is computed using the extended max-flow problem as explained in section 7.2.4.1.

Though there is a difference in the implementation of algorithms, the scenarios given below always discover the same routes in both simulation and analytical environments.

7.4.1 Comparison of Simulation and Analytical Environment

The same parameters shown in Table 7-7 are used for all the simulations. Table 7-7 compares both parameters used for the simulation and the analytical results.

Table 7-7 Parameters used for simulation and analytical environments

Parameters	Simulation	Analytical
Link Properties		
	802.11b (ad hoc mode)	Each link capacity is limited by 1 unit (a constraint given to the LP). Each node has infinite buffers.
	PHY mode is set to 1 Mbps	
	RTS/CTS enabled	
	Tx power 100 mW	
	Rx threshold -76dBm	
	Large packets are fragmented (threshold is set to 2304 bytes)	
	Buffer size 1024 Kbytes	
Propagation Model	Free Space (used in OPNET Simulator)	Free Space (used to compute the signal strength at the receiver)
Communication Ranges		
Transmission Range	600m	1 unit
Carrier Sensing Range	600m	1 unit
RDM Protocol		
Maximum number of routes used	2	2
Method of distribution	Splitting	Splitting
Selection of RDM paths	see section 3.1.2	see section 7.2.4.1
Node mobility	No (Hello disabled)	No
SP route	Path with the lowest hop count	
Simulation Details		
Simulation duration	400 sec	-
Application	CBR UDP data stream	-

In general, simulation results show a lower sustainable throughput than the analytically computed sustainable throughput with optimal scheduling. This is mainly due to the following reasons.

- There is no scheduling performed in the simulation. This means the contention, defined in the 802.11b MAC layer (CSMA/CA) can reduce the network

throughput further. Therefore, the simulated throughput is always below the analytical throughput.

- The analytical model assumes that there are no packet collisions, while this can happen in the simulation environment due to the hidden node problem. In the simulation, there are 2 possible causes for packet collisions due to the hidden node problem as shown in Figure 7-10. Assuming the CS range, R_{cs} is only up to the direct neighbor, in case (1), node n_p is a hidden node for node n_r. Therefore, the simultaneous transmission from node n_p and node n_r causes both packets to collide at node n_q. In case (2), a transmission form node n_r to node n_s, node n_p is a hidden node. This results in not receiving a packet at node n_q, while node n_s receiving the packet properly. For the unidirectional data transmission, data packets can be lost as in case (2). But, WLAN layer ACK packets can be lost as shown in either case. In order to avoid the hidden node problem, the simulations are run with RTS/CTS messages. Though the data packets are not lost due to hidden node, WLAN ACK and RTS/CTS packets can be lost, causing a degradation of the throughput due to retransmissions and finally dropping the data packet after several unsuccessful tries of RTS. The simulation throughput can further be reduced since some possible transmissions can be blocked (section 10.2.2 of Appendix II) with RTS/CTS messages.

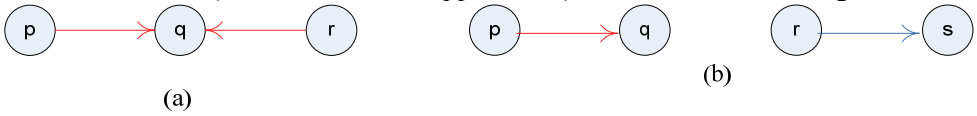


Figure 7-10 Hidden Node Problem – (a) Case 1 (b) Case 2

- The analytical model computes the throughput without considering the overhead of the headers at the different layers. In the simulation, the overhead of UDP, IP and MAC headers occupy part of the bandwidth. Moreover, the analytical model does not consider the bandwidth taken to exchange WLAN layer ACK packets and RTS/CTS control packets. The percentage of bandwidth taken for the above overheads w.r.t. to the available physical layer bandwidth is compared in Table 7-8. The values are taken from the 3x3 grid topology used in the simulation.

Table 7-8 Simulation results - % of throughput taken by data and control messages (3x3 grid topology)

Computation of the throughput		% of bandwidth taken w.r.t. 1 Mbps
Measured at the application layer, 2275 bytes x 20 packets	2275x20x8 = 364000 bps	36.4 %
Bandwidth taken by the headers (UDP header = 8 bytes, IP header = 20 bytes, MAC = 28 & PHY header = 24 bytes)	(8+20+28+24)x20x8 = 12800 bps	1.28%
Bandwidth taken to send control messages (RTS, CTS & WLAN-ACK)	608 bps	0.06%

7.4.1.1 Sustainable throughput with optimal and non-optimal scheduling

In order to additionally verify the analytical results compared to the simulation results, the analytical results are also taken by scheduling one link at a time, called "non-optimal scheduling". This means that each link is put into one IS. When using "non-optimal scheduling" for the 3x3 grid topology, as explained in section 7.2, 8 links in the FRDM paths are scheduled with 8 ISs. In this case, the sustainable throughput is computed giving an active time of 0.25 by using only one path. Table 7-9 compares the analytical and simulation throughput for the 3x3 grid topology. These results show that the simulation throughput lies between the analytical throughput with optimal scheduling and non-optimal scheduling.

Table 7-9 Analytical results vs simulation results (3x3 grid topology)

	Analytical		Simulation
	With Optimum scheduling	Non-optimal scheduling	Using 802.11b MAC
% Sustainable throughput	66.7	25	36.4

The sustainable throughput of the analytical results given below is computed for both optimum and non-optimum scheduling. When using RDM routes, the sustainable throughput with non-optimum scheduling is always computed by utilizing one path. When using one link at a time (non-optimal scheduling), the max-flow problem shows the use of SP (with scheduling one link at a time) gives higher throughput than using RDM paths simultaneously by scheduling one link at a time.

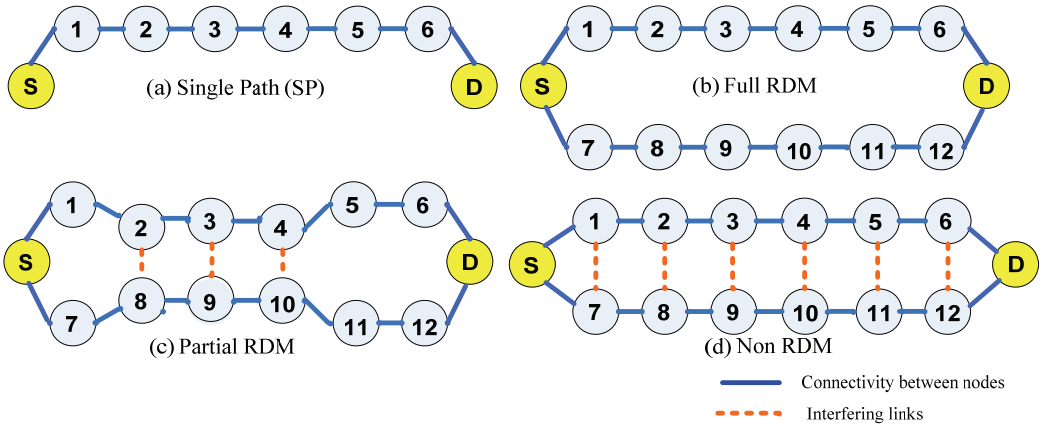


Figure 7-11 Different basic topologies

7.4.2 Basic Topologies

Figure 7-11 shows the use of SP and different types of RDM routes in a simple network, where S denotes the sender, n_S and D denotes the destination node, n_D. Dashed lines in the PRDM and NRDM scenarios show the interfering links.

Table 7-10 Basic topologies – ISs used in scheduling

Scenarios		Used Independent Sets	Active time fractions
Single Path	SP	IS_1.1: (S-1, 3-4, 6-D)	0.333
		IS_1.2: (1-2, 4-5)	0.333
		IS_1.3: (2-3, 5-6)	0.333
RDM Paths	FRDM	IS_2.1: (S-1, 11-12, 3-4, 6-D, 8-9)	0.333
		IS_2.2: (1-2, 10-11, 4-5, 7-8)	0.333
		IS_2.3: (S-7, 12-D, 2-3, 5-6, 9-10)	0.333
	PRDM	IS_3.1: (S-1, 11-12, 4-5, 8-9)	0.25
		IS_3.2: (1-2, 12-D, 5-6, 9-10)	0.25
		IS_3.3: (S-7, 10-11, 2-3, 5-6)	0.25
		IS_3.4: (11-12, 3-4, 6-D, 7-8)	0.25
	NRDM	IS_4.1: (S-1, 12-D, 4-5, 8-9)	0.25
		IS_4.2: (1-2, 5-6, 9-10)	0.25
		IS_4.3: (S-7, 10-11, 2-3, 6-D)	0.25
		IS_4.4: (11-12, 3-4, 7-8)	0.25

Table 7-10 shows how scheduling is done in the analytical model. In the SP scenario, all links are divided into 3 groups (ISs), and each group is assigned equal amounts of time, i.e. 1/3 of the total available time. In the FRDM scenario, there are 3 ISs consisting of links from both paths and the total available time is equally distributed to those 3 groups of links. This means that the path S-1-2-3-4-5-6-D is scheduled exactly in the same way in both scenarios. But there exists another path S-7-8-9-10-11-12-D in the FRDM scenario, which is just like a clone of the path S-1-2-3-4-5-6-D, and this is the reason that the sustainable throughput of this scenario has been doubled, in comparison to the SP (Table 7-11).

Table 7-11 compares the analytical sustainable throughput of different types of routes. This shows that the simultaneous use of FRDM together with optimized scheduling can double the sustainable throughput, compared to SP throughput. In the NRDM scenario, where each intermediate node in one path interferes with the one in the other path, there are 4 ISs that have been chosen for scheduling. Since the links are conflicting (i.e. interfering), more links cannot put it one IS like in FRDM case. The active time for one IS is given as 25%. This computes the throughput of one path as 25% and since 2 paths are used, the overall throughput can go up to 50%. PRDM scenario also gives 4 ISs, including link l_{11_12} in both IS_3.1 and IS_3.4. The sustainable throughput with non-optimal scheduling is same for all cases since it uses only a single path even with RDM routes as discussed in section 7.4.1.1.

The simulated sustainable throughput of unidirectional transmission in Table 7-11 always lies between the analytical throughputs with optimal and non-optimal scheduling. As discussed in section 7.4.1.1, simulation throughput is always lower than the analytical throughput with optimal scheduling due to no scheduling, contention delay,

overhead of headers and RTC/CTS control messages in the simulation environment. Simulations have also been done to evaluate the sustainable throughput for bidirectional UDP data transmission.

Table 7-11 Basic topologies - % of Sustainable throughput

Scenarios	Analytical		Simulation	
	Unidirectional		Uni-UDP	Bi-UDP
	Non-optimal scheduling	Optimal scheduling		
SP	14.28	33.3	23.1	8.96
FRDM	14.28	66.7	36.4	15.2
PRDM	14.28	50.0	28.8	5.44
NRDM	14.28	50.0	27.2	5.28

Compared to FRDM, simultaneous use of NRDM limits the sustainable throughput due to interference between paths. But, NRDM still performs better than SP in unidirectional communications. With bidirectional traffic, there are more contention delays in the network. This is due to the fact that the traffic arriving from the opposite directions may increase the probability of having collisions between links. The consequence is the exponentially increased contention delay. The performance gain of FRDM w.r.t. bidirectional communication is 1.7, whereas it is 1.6 for unidirectional communications. With the use of bidirectional communications for NRDM and PRDM, the results show that there is no performance gain w.r.t. SP. Since there are already many interfering links and additionally there is no scheduling in the simulations, the dramatically increased contention delay may degrade the sustainable throughput drastically, when using interfering routes simultaneously.

In summary, both analytical and simulation results show that sustainable throughput can be enhanced when using FRDM paths, for both unidirectional and bidirectional data transmissions. Furthermore, the simulation results verify that simultaneous use of FRDM routes increase the sustainable throughput, even without any optimal scheduling. Additional simulation results which are not detailed here show that the performance of NRDM scenarios degrades further with the use of bidirectional traffic and the increase of hop lengths.

7.4.3 Grid Topology

The 3x3, 4x4 and 5x5 grid networks as shown in Figure 7-12 have been evaluated to compute the sustainable throughput. There is no BTL in these networks. As in Figure 7-12 (a), the node located in the lower left corner is always set as the sender, and the node in the upper right corner is set to be the destination. It is same for all 3 scenarios that the most outer paths are discovered as RDM routes for data transmission due to having the least mutual interference. In the SP scenario, one of the outer paths is selected for the data transmission. Table 7-12 compares the simulation and analytical results.

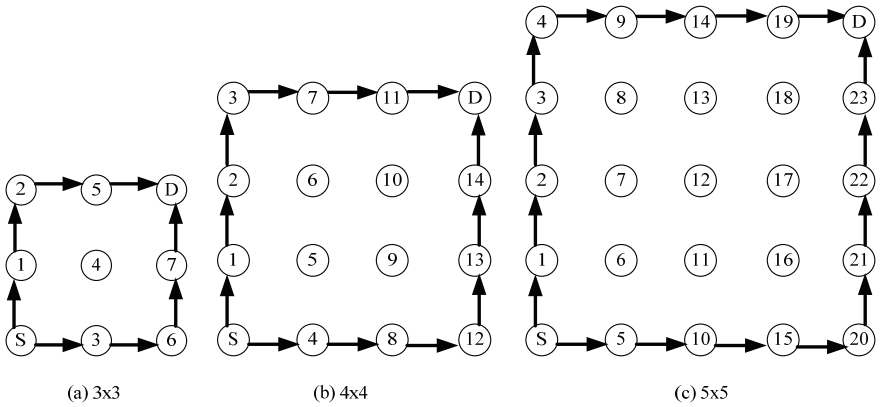


Figure 7-12: Grid networks of different sizes

Table 7-12 Grid topology with different sizes - % of Sustainable throughput

Scenarios		% of sustainable throughput		
		Analytical		Simulation
		Non-optimal scheduling	Optimal scheduling	
SP	3x3	25	33.3	26.0
	4x4	16.67	33.3	25.6
	5x5	12.5	33.3	23.1
FRDM	3x3	25	66.7	36.4
	4x4	16.67	66.7	36.4
	5x5	12.5	66.7	36.4

When only an SP is selected, the analytically computed throughput with optimal scheduling is same for all 3 grid networks, while the analytical throughput with non-optimal scheduling and the simulation throughput decrease with the increase of the hop length. This is due to every link experiencing more channel access contention from other links when the number of hops increases in the simulation.

Simulations give out a constant maximum throughput for FRDM scenarios, irrespective of the change of hop counts, since the paths are not overloaded (due to the distribution of packets among 2 non-interfering paths). These results also show that the simulation throughput lies between analytically computed throughputs with optimal and non-optimal scheduling.

7.4.4 String Topology

The scenarios discussed above only consider RDM paths with the same hop count and no BTL. Figure 7-13 shows a 17 node network, where RDM paths have different hop counts. Here, 8% of link capacity between node n_8 and node n_9 of the shortest path is utilized for a BTL. Without RDM routing, the SP is chosen as the path with the lowest hop count, i.e. the middle path. RDM routing avoids the selection of this as the primary path due to the existing BTL and also having higher number of interfering links with the upper and the lower paths. It selects only the upper and lower paths as FRDM paths. In

order to create NRDM paths, all three paths (i.e. upper, lower and the middle) have been used to distribute packets in this scenario.

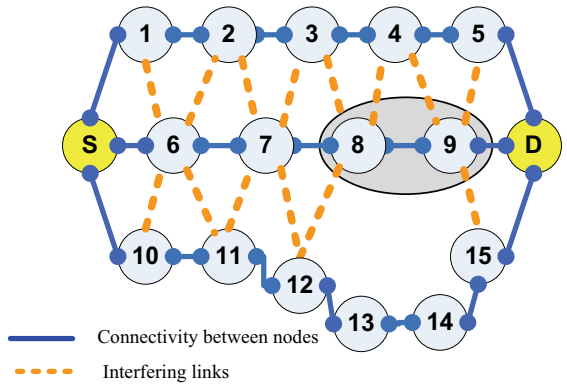


Figure 7-13 17 nodes string topology

Table 7-13 shows that the FRDM routes perform better than the SP for both simulation and analytical environments with optimal scheduling. The use of all three routes results in creating NRDM routes for this topology and it shows a degrading performance for the simulation results due to the mutual interference between 3 paths. On the other hand, analytical results of NRDM show relatively higher throughput than the analytical SP throughput. This is due to the distribution of traffic among three paths together with optimal scheduling by avoiding the simultaneous use of interfering links. Table 7-13 computes the performance gain of RDM routes w.r.t SP throughput for simulation results. It shows that FRDM always has a throughput gain when using real 802.11 based multi-hop ad hoc networks. In case of using the non-optimal scheduling in the analytical model, the FRDM scenario selects only the upper path while the NRDM scenario selects only the middle path for non-optimal scheduling.

Table 7-13 String topology - % of Sustainable throughput

Scenarios	Analytical		Simulation			
	Non-optimal scheduling	Optimal scheduling	Uni-directional	Perf. Gain w.r.t. SP	Bi-directional	Perf. Gain w.r.t. SP
SP (middle)	16.93	28.21	20.8	-	7.68	-
FRDM (Upper & Lower)	16.67	60.0	32.0	1.54	14.4	1.875
NRDM (Upper, Lower & middle)	16.93	56.0	15.2	0.73	5.92	0.77

7.4.5 5x5 Grid Topology with Background Traffic Load

This scenario consists of 25 nodes. The evaluated routing paths between the sender (n_S) and destination (n_D) are shown in Figure 7-14. A unidirectional BTL is added to links connected to the 6 nodes of: n_6 & n_7, n_8 & n_9 and n_{10} & n_{11} (nodes in the dark areas of Figure 7-14) as BTL.

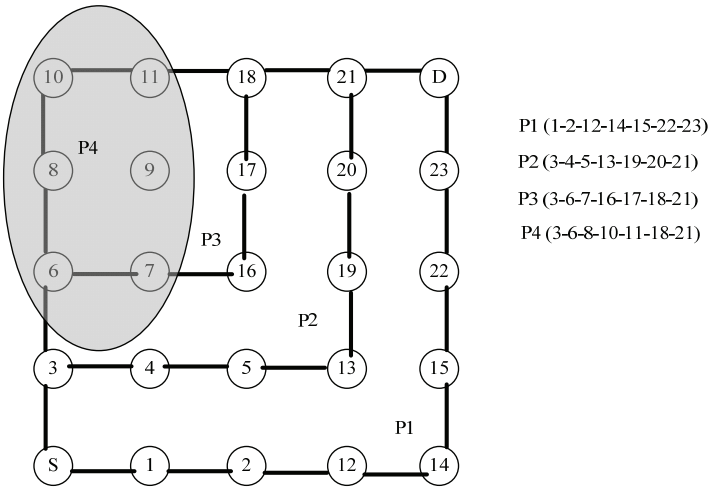


Figure 7-14: Possible routes in a 5x5 grid network with consideration of BTL

This scenario is used to compare the sustainable throughput with the increase of the BTL. When using the RDM routes, P_1 is selected as the primary path (with zero BTL and having least interference). Computation of mutual interference w.r.t. P_1 gives the highest mutual interference for path P_2 (NRDM routes) and zero mutual interference for paths P_3 and P_4 (FRDM routes). This scenario represents 3 different types of routing.

- SP – P_1
- NRDM – (P_1 and P_2)
- FRDM – (P_1 and P_3) or (P_1 and P_4)

The effect of BTL is considered in the analytical model as explained in section 7.2.4.2. In the simulation, BTL is generated using a unidirectional UDP communication by varying the data rate. Figure 7-15 shows that the analytical sustainable throughput of FRDM is doubled compared to SP when the link l_{67} is carrying less BTL (< 20%). In this scenario, NRDM paths do not carry any BTL, but the performance is degrading with the increase of the BTL on the link l_{67}. When using FRDM, the primary path (P_1) is always used to send the data at its maximum rate (i.e. 33.33% without any BTL in the P_1). For example, when the BTL is 40% of the link capacity, sustainable throughput of

FRDM is computed as 43.33%. This is achieved by sending 33.33% on path P_1 and 10% on path P_3. The link l_{67} on P_3 has been used by BTL and therefore available link capacity is only 60%. With optimum scheduling, 10% of the available capacity has been used in P_3 for the current communication.

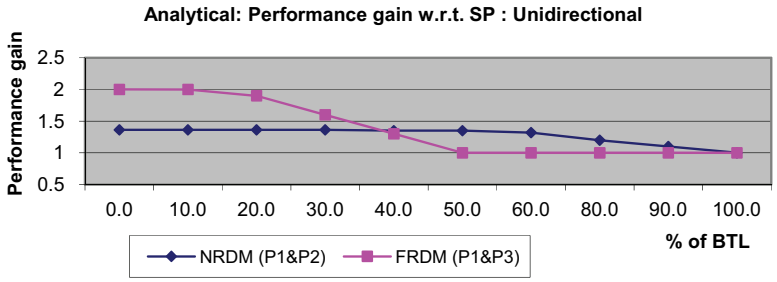


Figure 7-15 Analytical results - Performance gain vs % of BTL

Simulation results for unidirectional UDP as shown in Figure 7-16, compare the performance gain with and without enabling RTS/CTS messages. This shows that both, for NRDM and FRDM the performance degrades with the use of RTS/CTS messages. Use of RTS/CTS increases blocking of nodes due to exposed node problem. The throughput degrades as more nodes are blocked, when there exist more interfering links as in NRDM. Though, there are no interfering links in FRDM, use of RTS/CTS reduces the sustainable throughput due to BTL used in P_3 of this scenario. These results show that when the links are congested with a BTL beyond 8% of the link capacity, there is no performance gain that can be achieved when using FRDM for unidirectional communications.

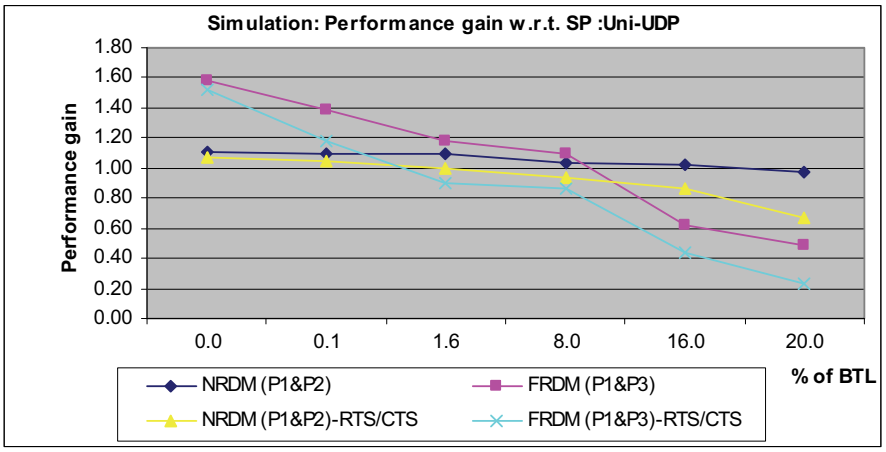


Figure 7-16 Simulation results (Uni UDP) - Performance gain vs % of BTL

Figure 7-17 shows the performance gain of FRDM and NRDM routes when using bidirectional UDP data transmissions. These results also show that RTS/CTS degrade the performance since the blocking probability caused by the exposed node problem is much higher when using bidirectional transmissions (section 10.2.2). In summary, this shows that FRDM can gain much higher performance (a factor of 4 for bidirectional and 1.6 for unidirectional) with very low BTL. The simultaneous use of FRDM routes show performance improvements with a BTL of 8% or less (when using RTS/CTS) and 16% or less (without RTS/CTS).

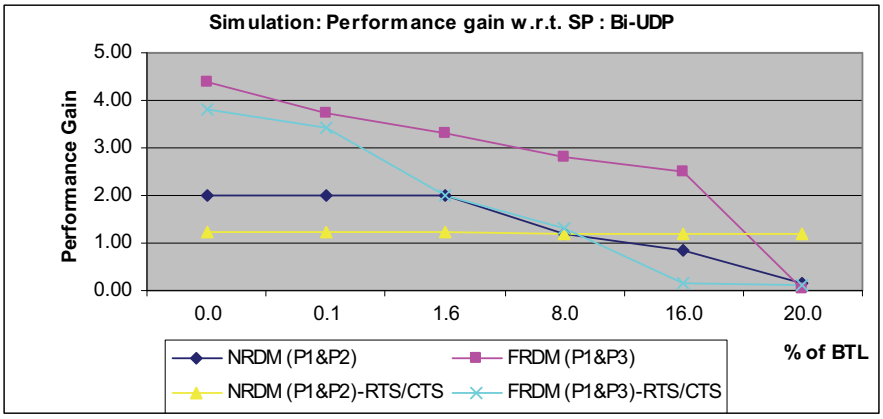


Figure 7-17 Simulation results (Bi UDP) - Performance gain vs % of BTL

7.5 Conclusions

This chapter has discussed an analytical model to assess the interference aware multipath routing scheme called RDM routes. Analytical results are compared to simulation results by computing the sustainable throughput of different network topologies. Compared to previous work of analyzing network throughput considering the impact of interference, this work pioneers determination of multiple routes by considering two factors: assessing the mutual interference between paths and background traffic of the paths [104, 105]. Previous work has not compared analytical results with simulation results taken from a real implementation of interference aware routing. This work gives a detailed analysis of the behavior of interference aware multipath routing considering both unidirectional and bidirectional communications.

7.5.1 Comparison with Simulation results

In the SP scenario, all data packets are transferred over the shortest path, while in other scenarios, bandwidth available for the sender is more due to the distribution of data among two paths. The results show that the use of 2 paths simultaneously by balancing the load gives a higher throughput, if paths are not interfering with each other and not carrying any BTLs. This work analyses how mutual interference between paths and the existing BTL of a path are affecting the degradation of the sustainable throughput.

Analytical results show that simultaneous use of RDM routes performs better than SP. It gives the maximum sustainable throughput when using FRDM routes. The sustainable throughput is degraded with increasing BTL. However, for the evaluated scenarios, PRDM and NRDM routes also perform better than SP due to the optimal scheduling used in the analytical model. The scheduling is used to make sure that interfering links are not active simultaneously.

Simulation results also show that FRDM routes always perform better than SP, NRDM and PRDM. If SP is congested or RDM paths have more mutual interference (as in Figure 7-13), sustainable network throughput degrades. The degradation of the sustainable throughput is more when using bidirectional data transmissions since the data traffic originates as nodes from both directions introducing more contention between links. The simulation results prove that FRDM routes perform better than SP even without any optimal scheduling as used in the analytical model. The performance gain that can be achieved with FRDM routes is much higher with bidirectional data transmissions, because FRDM alleviates the contention by distributing packets among non-interfering links. If FRDM paths themselves have a BTL, the use of FRDM routes with a higher BTL results in degrading the performance.

Simulation results show that when using bidirectional UDP data, the sustainable throughput is degraded drastically. In the analytical model, computation of sustainable throughput using the max-flow problem is limited only to the unidirectional transmissions. So far, no constraints have been included to consider bidirectional transmissions to the max-flow problem due to the complexity of solving the linear program. However, with the help of ISs (computed assuming bidirectional transmissions in each link), it could be easier to explain why the throughput using bidirectional transmissions is much lower than the one computed by using unidirectional traffic. Bidirectional transmissions limit the number of links that can be included into one IS.

7.5.2 Computational Cost of the Analytical Model

Discovery of ISs are an NP hard problem. Detailed analysis of the computational time taken by each step shows that more computational time is consumed to discover all ISs. Effort given in Table 7-14 is defined as one entire execution of the algorithm to discover one IS (see *Process B* of Figure 9-2 of Appendix I). Table 7-14 shows that the actual program running time is dependent on the total number of efforts. The number of efforts indicates the total number of discovered ISs, including the ones which have been discarded. The results are obtained using a laptop with Celeron mobile 1.4 GHz processor and 768M of RAM.

Table 7-14 Computational Cost

Scenario	Efforts	Program running time (in millisecond)
3x3 (9 nodes)	18	17
4x4 (16 nodes)	96	209
5x5 (25 nodes)	477	5891

7.5.3 Enhancements to the Analytical Model

Validation of Analytical and Simulation Results: This analytical model assumes that there exists a central entity that does the global scheduling and thereby not considers the real behavior of 802.11 MAC such as contention delay and packet collisions. Therefore, the simulation results do not exactly compare with the analytical results. This validation can further be improved either by changing the analytical model to consider 802.11 MAC behavior or by taking the simulation results with a link layer that does the global scheduling (e.g. TDMA) instead of using 802.11b MAC. For the first validation, there exists some research modeling the single hop behavior of 802.11 MAC [106, 107]. However, this gets more complicated when extending these algorithms for multi-hop connections in terms of computational time, when considering the interference from other hops.

Mobility: In this model, a conflict graph is generated assuming all the nodes are stationary. Mobility of the nodes can also be included by regenerating the connectivity and the conflict graphs for new topologies. When the position of a node changes, a change in the topology (adding or removing the links in the connectivity graph) has to be modeled. This leads to the change in conflict graph and reevaluation of new RDM paths. In this process, the most difficult task is to generate the conflict graph based on the physical interference model, because extreme node movements can affect the SNR of all the nodes in the vicinity. Generation of the conflict graph and discovery of ISs consume more computational time. Therefore, in order to save the computation time, a new conflict graph should be built based on the previous topology considering only the changes of movement of nodes.

8. Conclusion and Outlook

Multipath routing is an improvement to single path routing to provide backup paths in case of path failures to prevent further route discoveries and to distribute flows (i.e. application data) to increase the effective bandwidth. In this thesis, the use of multipath routing has been enhanced by discovering non-interfering and least congested routing paths and using them simultaneously to distribute application flows based on different distribution methods.

A fundamental difference of wireless networks from wired networks is the mutual interference between links/paths located in proximity to each other. When multiple routing paths are interfering with each other, these paths cannot be operated simultaneously as in the case of a shared medium such as the IEEE 802.11 technology. The quality of transmissions may be degraded due to interference, even though the paths are node disjoint. There are a few proposed metrics to measure the independence between the links of different paths as discussed in Chapter 2. This thesis proposes a novel mechanism to discover multiple paths without or with minimum mutual interferences.

The new protocol developed in this thesis is named Radio Disjoint Multipath (RDM) routing. It selects multiple routing paths by considering the mutual interference between paths together with the existing traffic load in a path. The functions of the RDM protocol are explained elaborating on the 5 major processes involved in setting up multiple paths; path discovery, path selection criteria, flow distribution criteria, path maintenance and dynamic path evaluation. The RDM protocol does not require any modifications to link layer technologies or the applications. The feasibility of the proposed methods is proved by implementing the RDM protocol in the OPNET simulator. In this thesis, two algorithms are proposed to compute the existing BTL of a path and to compute the mutual interference between paths by maintaining INLs. These algorithms are validated through simulations. The simulation results prove that the computation of BTL which is done using the proposed weighted average algorithm is accurate. The accuracy of the computation of the INL depends on the propagation of RREQ messages in a network. Since the probability of losing RREQ messages is higher with the increase of the BTL and a higher number of nodes in the network, the INL is not computed accurately for the scenarios with higher BTLs. But, it is observed that more than 94% of the overall INLs are discovered successfully for these scenarios. This thesis further discusses how to reduce the number of lost RREQ messages. This thesis also discusses the feasibility of implementing the RDM protocol in real wireless multi-hop ad hoc environments.

Today's Internet based applications mainly use TCP and UDP protocols as the transport layer protocol. These protocols behave differently in wireless multi-hop ad hoc networks. This thesis provides different distribution mechanisms to enhance the performance of existing applications by using the RDM paths simultaneously. It introduces a novel mechanism to distribute multiple flows and packets of a single flow based on the PL, which is computed considering the BTL and the mutual interference between paths. Previous work analyses the performance of applications with SUM routing when selecting routes without considering the effects of mutual interference. This thesis considers the different combinations of routes to analyze the application performance.

The evaluation of results is done considering the non-interfering RDM routing paths (FRDM routes), the interfering RDM routing paths (NRDM routes) and single path routing. When using RDM routes, two distribution methods, viz., Multiple Flow (MF) and Single Flow (SF) distribution methods are considered. The MF distribution distributes individual flows over maximal two radio disjoint paths according to the remaining bandwidth, computed based on the PL. Simulation results show there is a significant improvement when using FRDM routes with MF distribution compared to the performance of SP and the NRDM routing paths. This is mainly due to alleviating the congestion on one path by distributing individual flows and also using non-interfering paths simultaneously. The use of SUM routing with the MF distribution in mobile scenarios show much better performance compared to the SP mainly due to the use of 2 paths which causes less route discoveries. The simultaneous use of interfering routes (NRDM) degrades the performance drastically, especially for TCP traffic. When transmitting TCP traffic on one path while the other path is carrying UDP traffic, the nodes carrying the UDP traffic capture the channel. Therefore, the nodes carrying TCP traffic have to wait too long and finally drop the TCP packets due to the capture of the channel by the other path. This does not happen when using the FRDM routes simultaneously since both paths are not interfering with each other. The SF distribution uses the round robin distribution to split packets. The distribution ratio is computed based on the PL, putting more packets to the path with less PL. The use of FRDM routes with an SF distribution also outperforms the SP and the NRDM performance, for the scenarios where the SP is heavily congested with the BTL and the discovered FRDM routes are having identical properties. Even though the RDM paths discovered in the mobile scenarios do not have identical properties, the splitting of both TCP based and UDP based packets show the improvement in the performance. The major contribution for the performance gain causes the use of active routes until the expiry of both routing paths. This results in considerably less route discoveries when using RDM routes simultaneously in mobile scenarios.

Mobile ad hoc networks (MANET) are applied in many emergency scenarios, e.g. fire-fighting coordination. This thesis also discusses the applicability of the RDM routing for emergency scenarios. It proposes an approach to make communications more reliable using the concept of packet replication and transmitting them over RDM paths. In contrast to balancing the load among the RDM routing, replicating each data packet among the RDM paths can increase the reliability of the communication. Although

replication causes higher congestion in the network, it enhances the reliability, especially in adverse propagation environments such as fire-fighting. An applicable scenario for fire-fighting is introduced summarizing the feasibility studies done to prove the suitability of using wireless technologies for fire-fighting applications.

An analytical model is proposed to determine the RDM routing paths in a given wireless multi-hop network by modeling the mutual interference between paths together with BTL. Analytical results are compared with simulation results by computing the sustainable throughput of different network topologies. Compared to previous work of analyzing network throughput considering the impact of interference, this work pioneers the determination of multiple routes by considering two factors: computing the mutual interference between paths and existing BTL of the paths. Previous work has not compared analytical results with simulation results taken from a real implementation of interference aware routing. The analytical results shows the use of multipath routing always enhances sustainable throughput of RDM routes compared to the SP routing with the use of optimal scheduling.

The work done in this thesis can be applied to enhance the application performance in different kinds of wireless multi-hop ad hoc networks such as Mobile Ad hoc NETworks (MANET), Wireless Sensor Networks (WSN) and Wireless Mesh Networks (WMN), by discovering the RDM routes and using them simultaneously.

Work in this area of research is expected to continue in the future due to its importance in the previously mentioned wireless networks. Major challenges will be to enhance the route evaluation process for networks with mobility. For example, parameters such as how often the triggering of route evaluation and route re-discoveries should be initiated have to be optimized for different speeds of mobility. Further, the proposed distribution algorithms should be enhanced to trigger and select the best algorithm to be used based on the scenarios.

9. Appendix I – Implementation Details

This appendix provides some of the flow charts of the implementations done in this thesis work for further clarifications.

9.1.1 Analytical model

This section provides an overview to the main classes and the flow chart of the implementation of the computation used in the analytical model explained in section 7.2.

The class *NetworkNode* has 3 main attributes of *node identification*, *geometric coordinates* and *connections to neighboring nodes*. The *node identification* is used to distinguish different instances of the class *NetworkNode*. It is a non-negative integer value, and is unique to each instance of this class. The *geometric coordinates* are a pair of float numbers, and they give the location information of a specific node in a two dimensional space. Each node is allowed to have at most 4 connections to its neighbors. The connections are named as *up*, *down*, *left* and *right*.

The class *NetworkLink* contains mainly 3 attributes, namely the *link name*, the *starting node* and the corresponding *ending node*. The *link name* is used to identify a link uniquely from the other links. It is implemented as a *string*, and is unique for each single link. The starting and the ending nodes show from whom is the link originated, and to whom is it pointed.

The class *NetworkHandler* is designed as a central entity, which can finely control and carefully schedule the transmissions inside a given wireless ad hoc network. All the algorithms and storage elements have been implemented in this class.

Figure 9-2 shows the algorithm implemented to discover all possible independent sets in a given link of the network.

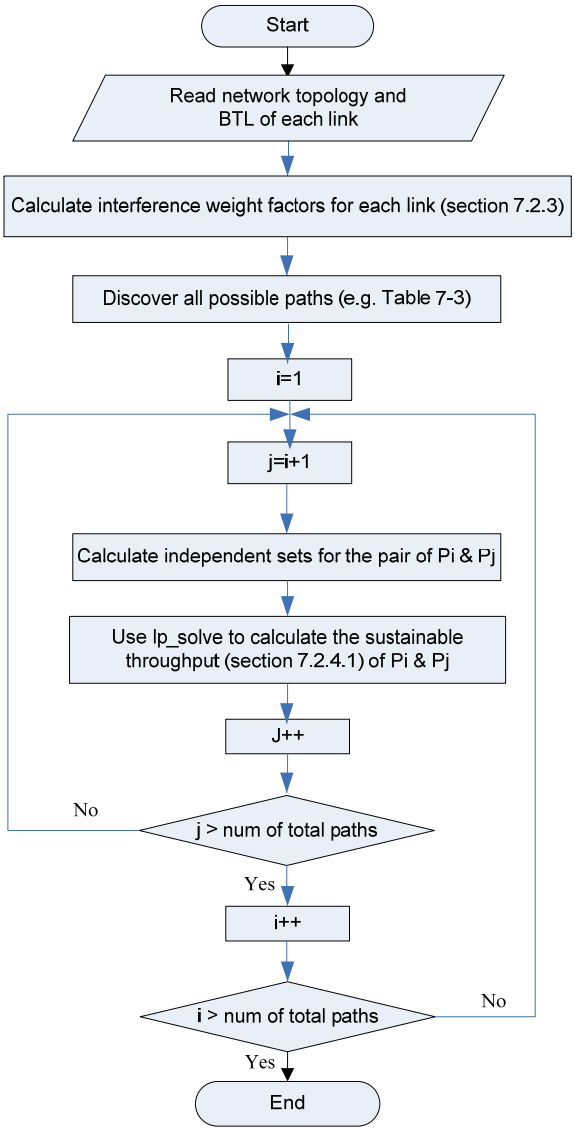


Figure 9-1 Flow chart of the implementation of the analytical mode

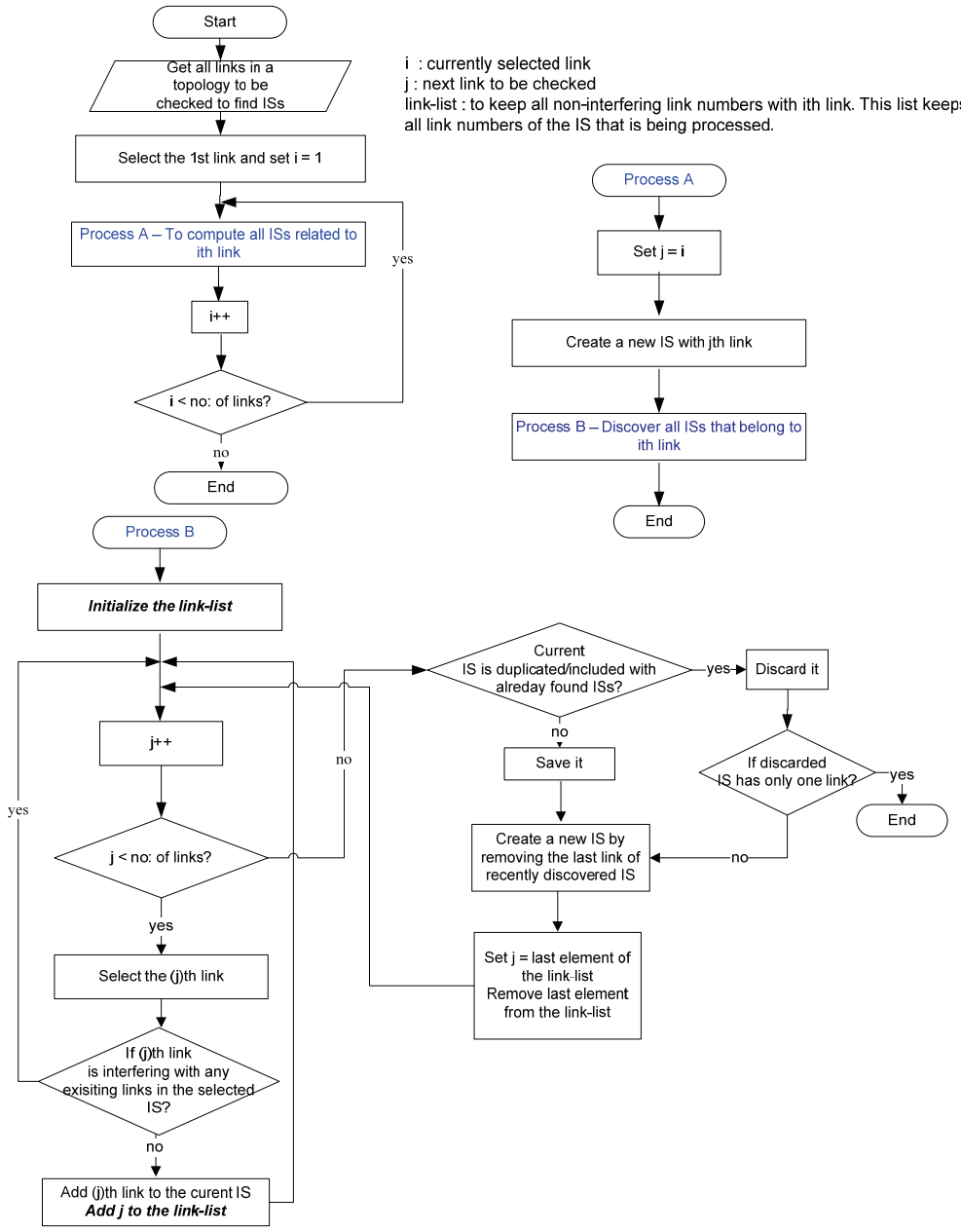


Figure 9-2 Computation of all ISs in a given network topology

9.1.2 Packet Replication and Discarding of Redundant Packets

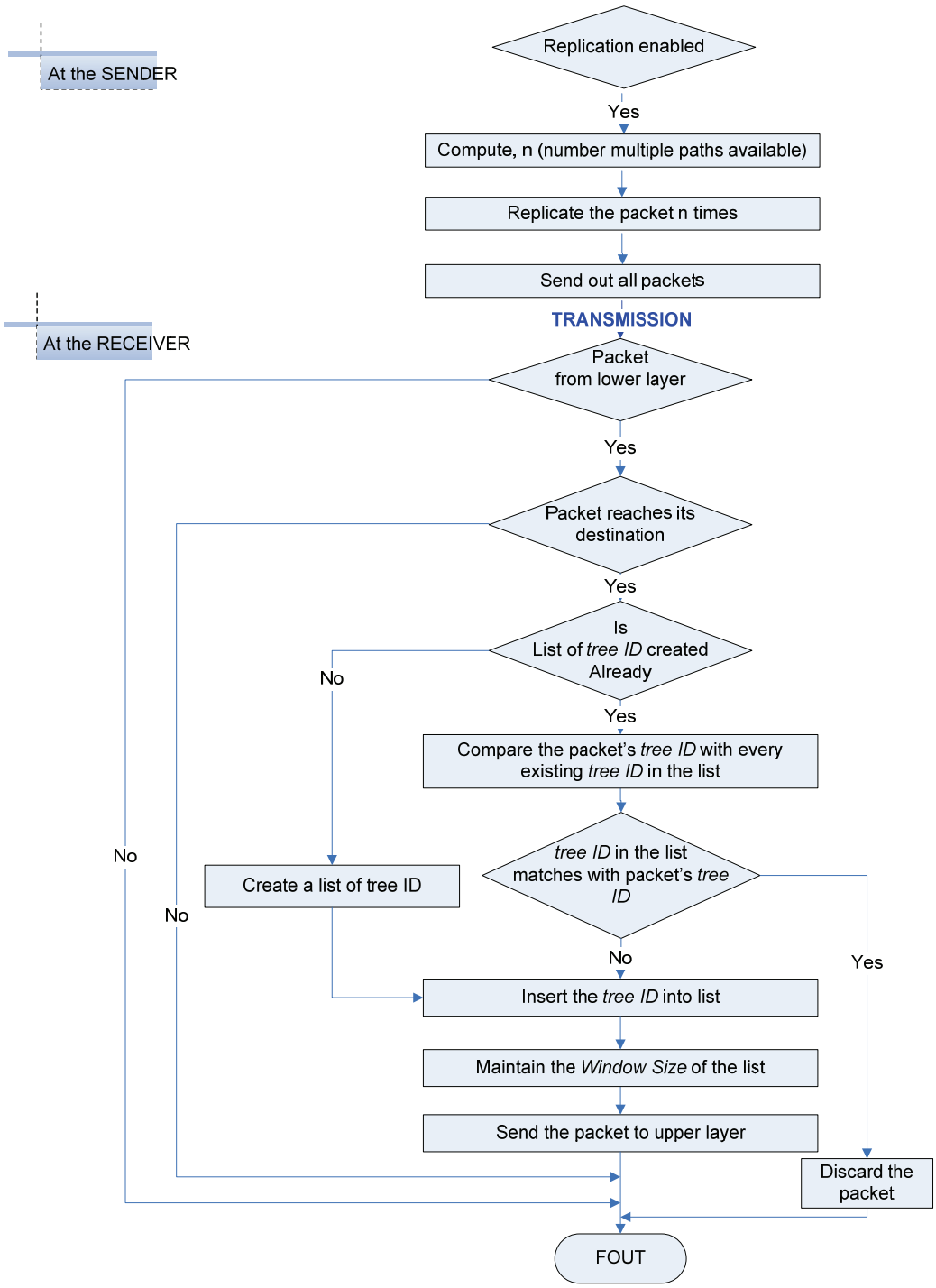


Figure 9-3 Discarding of redundant packets at the receiver

10. Appendix II – Detailed Glossary

10.1 Transmission Ranges used in IEEE 802.11

Three types of communication ranges exists in IEEE 802.11 wireless technologies, namely the *transmission range* R_{Tx}, the *carrier sensing range* R_{cs} and the *interference range* R_i.

10.1.1 Transmission Range

The transmission range R_{Tx} represents the range within which a packet is successfully received if there is no interference from the other nodes, or the interference is not strong enough to disturb the data transmission. The transmission range is mainly determined by the transmission power, the radio propagation properties (i.e. attenuation) between the source and the receiver and the receiver sensitivity.

10.1.1.1 Carrier Sensing Range

The carrier sensing range R_{cs} is the range within which a transmitter triggers carrier sense detection. This is usually determined by the carrier detection sensitivity. In IEEE 802.11 MAC, a transmitter only starts a transmission when it senses the media free, and this is done by the Clear Channel Assessment (CCA) function [48].

R_{cs} of a node is independent of the PHY mode it uses, because the carrier sensing mechanism always operates at 1 Mbps. Therefore the carrier sensing range R_{cs} is equal to the transmission range R_{Tx}, when using PHY mode of 1 Mbps.

10.1.1.2 Interference Range

The interference range R_i is the range within which nodes in receive mode is "interfered with" by other transmitters. This range is usually determined by the transmission power of the other senders, the geometrical location of nodes and the used PHY mode.

Unlike R_{Tx} and R_{cs} which are defined for the transmitters, R_i is defined from the receiver's point of view. It is not possible to define an exact distance of R_i. The interference range R_i is meaningful only if it is defined specifically for a pair of nodes. An example is given in Figure 10-1 for further understanding.

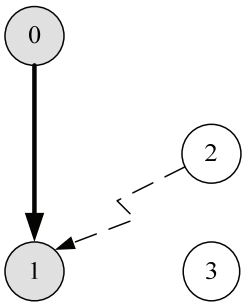


Figure 10-1 Interference range of node n_1

In this example the node, n_0 is sending data to the node, n_1. The nodes, n_2 and n_3 are located closer to n_1 and are sending data to anonymous nodes outside the figure. Since the node n_2 's transmission power is strong enough, n_1 is interfered by it. Although n_3 locates even closer to n_1, it cannot influence n_1 receiving data from n_0, because it is not sufficiently strong enough. In this case, it can be said that the interference range between n_1 and n_2 is larger than the one between n_1 and n_3.

10.2 Packet Drops in IEEE 802.11

This section details the reasons for the packet losses that are identified during the analysis of the simulations done in this work.

10.2.1 Hidden Node Problem

A hidden node problem occurs in wireless multi-hop ad hoc networks, when multiple nodes try to access the same node. This happens when two transmitting nodes are located physically beyond each other's carrier sensing range. The nodes which cannot detect each others' carrier sensing range may start transmitting at the same time to a common receiving node. In this case, packets can collide at the receiving node.

When using RDM paths simultaneously and if the packets are forwarded via two paths, which have identical properties, packets can collide easily at the destination node. This happens in the basic topology scenario (Figure 5-1). Figure 10-2 explains the TCP-Data packet collision that occurs at the source due to the hidden node problem. Assuming TCP-Data segments are arriving via the upper and lower paths of the basic topology scenario, Figure 10-2 shows how two TCP-Data segments are transmitted by the node 1 and node 7 to the source node. They are transmitted at the same time as they arrive to the node 1 and the node 7 almost at the same time. The original transmission fails due to the collisions of the two packets at the source node. Then, each node tries to transmit the TCP-Data segment until the retransmission retry limit of 7. Each attempt starts after a back-off period. The exponential back-off used in WLAN after unsuccessful transmission makes sure that both the node 1 and node 7 are not transmitting at the same time. Although they are not transmitting at the same time, two TCP-Data

segments still collide since transmission delay of TCP-Data segment (~ 18 to 20 ms) is much higher than the time difference between the starting time at the node 1 and the node 3. Therefore, all 7 attempts become unsuccessful and both TCP-Data segments are dropped by the node 1 and node 7 in this scenario. Retransmission attempts are successful when using the audio and the video flows since the size of the data packets are smaller.

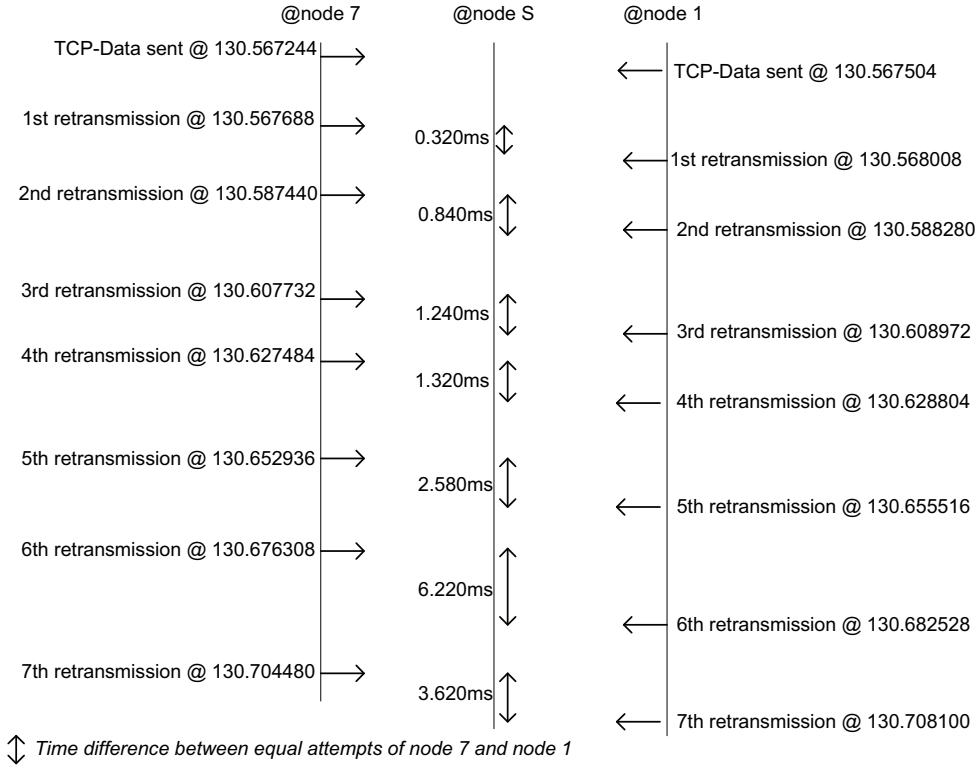


Figure 10-2 Packet collisions due to the Hidden Node Problem

10.2.2 RTS/CTS Handshake

In order to avoid the hidden node problem, RTS/CTS handshake is used in WLAN. RTS/CTS handshake makes sure that the receiving node does not allow receiving data from multiple nodes at the same time. However, the use of RTS/CTS handshake can delay the data transmission due to following.

- RTS/CTS handshake requires more time to exchange the control data.
- Use of RTS/CTS can block the other possibly successful transmissions unnecessarily.

Figure 10-3 shows an example of possible transmission in a Grid topology (Figure 5-3) when using NRDM routes. Assuming the node 23 is transmitting a packet to the node

22, any neighboring nodes which hears the RTS message from the node 23 (*node 20 and node D*) or the CTS message from node 22 (*node 19 and node 15*) cannot transmit or receive any data packets. However, the following transmissions are still possible without any interference to the on-going communication between the node 23 and the node 22.

- Node 14 can transmit to node 15
- Node 13 can transmit to node 19

But the above communications are blocked due to the RTS/CTS exchange between the node 23 and node 22. Further, as shown in Figure 10-3, only the following transmissions can occur together with the transmissions between the node 23 and the node 22.

- Possibility 1: "node 13 to node 5 or vice versa" and "node 3 to node S or vice versa"
- Possibility 2: "node 14 to node 12 or vice versa" and "node 1 to node S or vice versa"

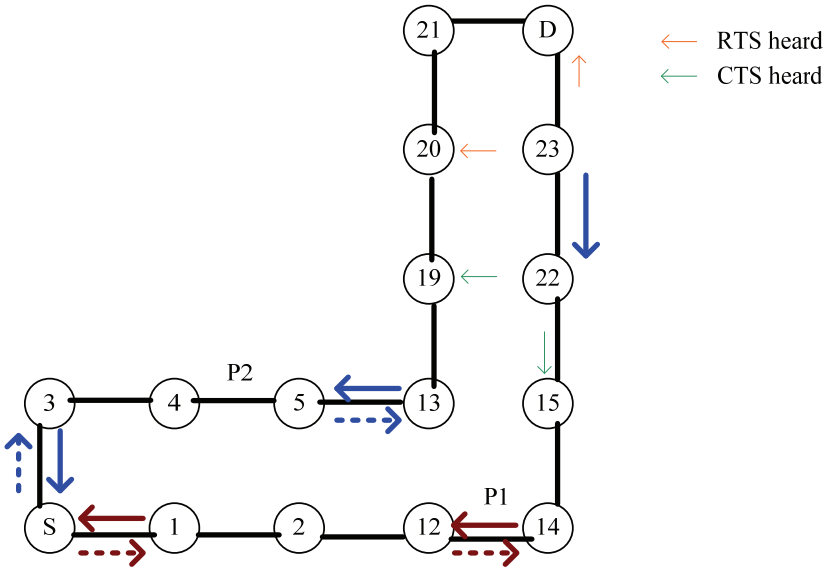


Figure 10-3 Possible transmission when RTS/CTS enabled

The use of RTS/CTS handshake can block more nodes transmitting at the same time. More and more nodes can prevent transmitting/receiving data if the nodes in two paths are interfering with each other. This situation favors successful nodes to send more data creating the capture condition in WLAN. The capture condition in WLAN causes unsuccessful nodes (blocking nodes) to drop packets due to unsuccessful handshakes, done through RTS/CTS messages.

10.2.2.1 Unsuccessful RTS/CTS handshake

This section details how a loss of a packet occurs in WLAN, when RTS/CTS handshake is enabled. This is investigated by tracking the time stamps of each transmission of a data packet. Figure 10-4 shows the timestamps recorded at the transport layer in

seconds, when starting a file download from the node D. For the simplicity of explanations, two digit numbers are used to identify a TCP-Data segment. This scenario represents a SP network in which only the immediate neighbors are within their transmission range. The following legends are used in Figure 10-4, Figure 10-5 and Figure 10-6.

- *TData[nn]*: TCP-Data segment which is tracked at the transport layer. The number "*nn*" refers to the segment number.
- *TAck[nn]*: TCP-ACK packet which is tracked at the transport layer. The number "nn" refers to the segment number.
- *WTData[nn] & WTAck[nn]*: Both TCP-Data segment and TCP-ACK which are tracked at the WLAN MAC layer. If these packets are marked by an underline, they refer to the packets that are transmitted by the WLAN layer.
- *WUDP:* UDP packets which are tracked at the WLAN layer.

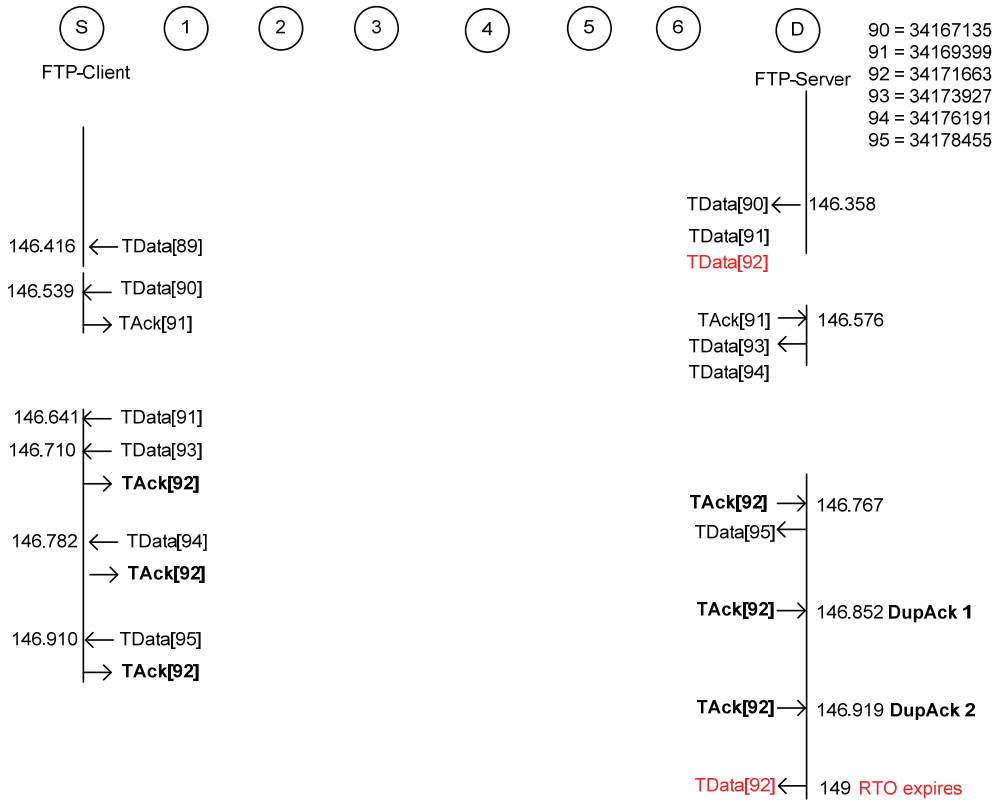


Figure 10-4 An example of a TCP timeout – SP scenario

Figure 10-4 shows that the 92nd TCP-Data segment is dropped during the transmission between the nodes D and node S. Therefore, the FTP client sends the TCP-ACK for the

92nd segment for each receipt of the subsequent TCP-Data segments. In this case, the FTP client does not receive enough subsequent TCP-Data segments to send the 3rd DupAck. Therefore, the FTP server detects the loss of the 92nd TCP-Data after the expiry of the TCP timeout.

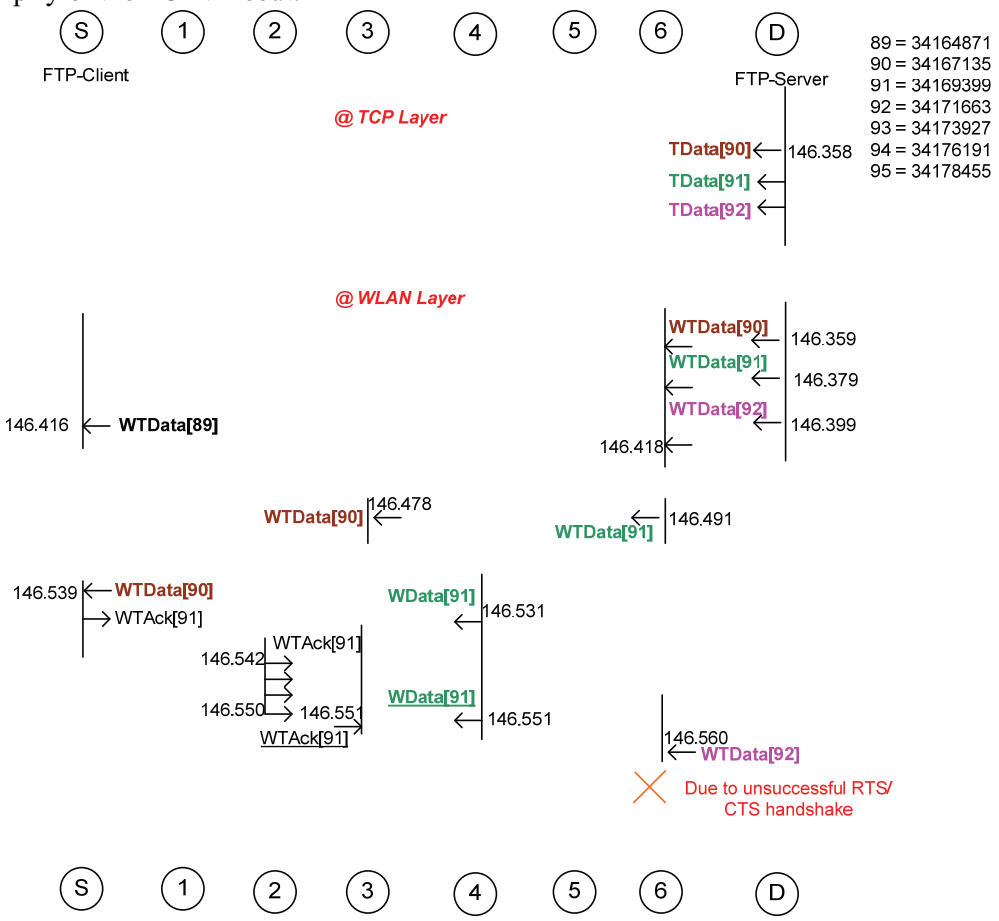


Figure 10-5 An example of a TCP-Data packet drops at the WLAN

Figure 10-5 shows how the TCP packets shown in Figure 10-4 are propagated at the WLAN layer in detail. The node 6 receives three TCP-Data segments at once from the node D. The node 6 tries to transmit 3 TCP-Data segments one after the other. As shown, 90th TCP-Data segment is transmitted successfully to the node S. The 91st TCP-Data segment has to be retransmitted twice (by the node 4) due to a collision with the TCP-ACK packet. During this time, the node 6 is trying to send the RTS message to the node 5 to send the 92nd TCP-Data packet. But, the node 5 is not able to send the CTS packet back to the node 6, since it has to be quiet during the transmission between the node 4 and the node 3. Therefore, the node 6 has to drop the 92nd TCP-Data segment after attempting 4 times to send the RTS packet to the node 5.

In summary, data packets might be dropped by the WLAN due to the unsuccessful RTS/CTS handshakes. This happens under the following conditions.

- *Congestion*: As shown in Figure 10-5, the node 6 drops the 3rd TCP-Data segment since it has to wait a longer time to send the data. When using a single flow and one node receives multiple data packets at once to be transmitted, RTS/CTS handshake may fail.
- *Capture Effect of WLAN*: When using multiple flows, the flows which send more data may have a higher probability of capturing the channel. For example, when downloading a file via FTP while sending a UDP flow on the same path, the UDP flow utilizes more bandwidth since UDP does not use any congestion control like in TCP. Therefore, the probability of TCP packets being lost and the chances of the UDP packets capturing the medium are much higher causing more TCP packets to be dropped at the WLAN layer. Similarly, when transmitting multiple TCP connections over the same path, the connections which send more data may have a higher probability of capturing the channel. An example of the capturing of the medium by the audio packets is shown in Figure 10-6. Due to the transmission of multiple audio packets (WUDP), the node 5 cannot send the CTS back to the node 6. Therefore the node 6 has to drop the 62nd TCP-Data segment.

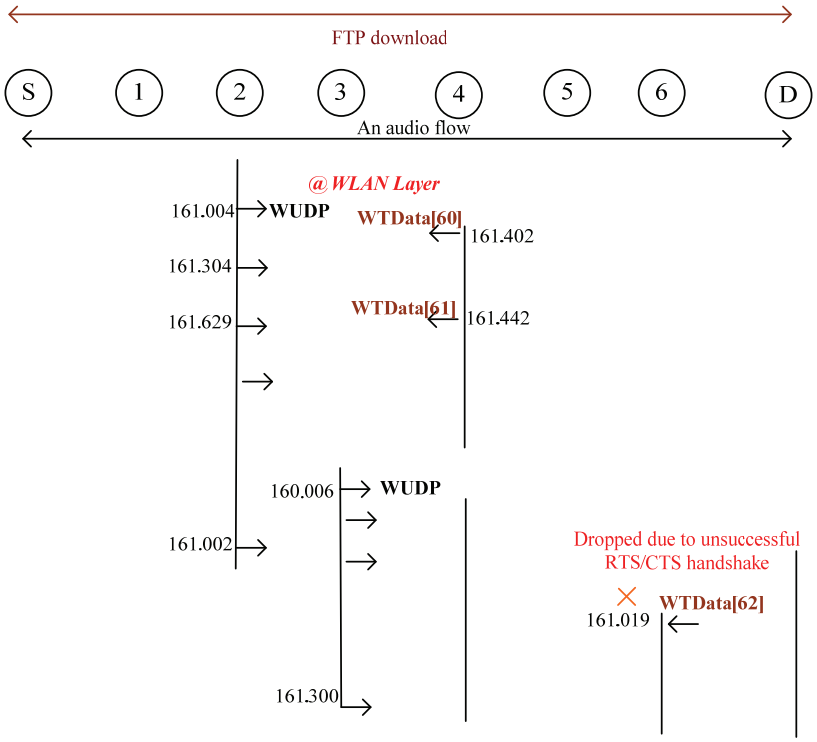


Figure 10-6 An example of a capturing media by audio packets

10.3 Effect of Maximum BTL in a node

This section details how BTL affects the performance of the application. This is evaluated in a SP scenario by applying different amounts of BTL as shown in Figure 10-7. In each scenario, the sustainable throughput is measured when sending a bi-directional UDP stream between the node S and the node D.

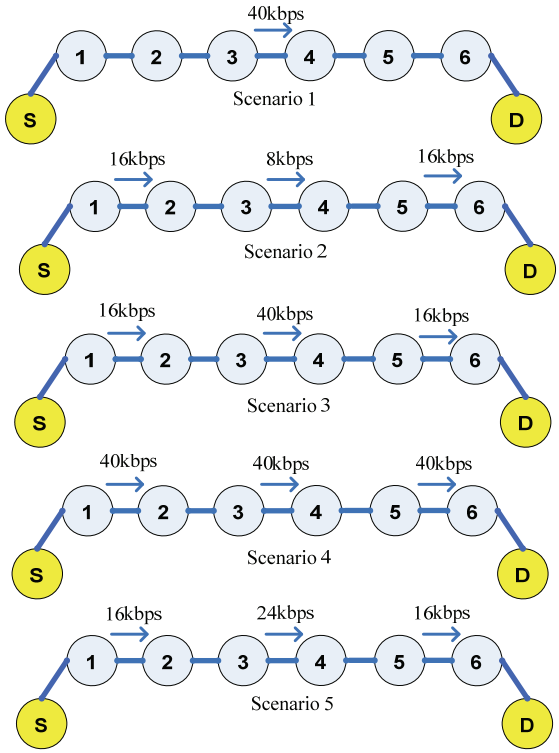


Figure 10-7 SP scenario with different amount of BTL used

Table 10-1 Sustainable throughput vs. BTL

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
BTL – MAX (bps)	40000	16000	40000	40000	24000
BTL – Accumulated (bps)	40000	40000	72000	120000	56000
Average BTL (bps) (BTLaccu/ number of nodes)	20000	6666.66	12000	20000	9333.33
Sustainable Throughput (bps)	51200bps	54400bps	50560bps	35200bps	52800bps
Sustainable Throughput (%)	5.12%	5.44%	5.05%	3.5%	5.28%

Table 10-1 shows that how sustainable throughput changes with the change of the BTL. Comparing scenario 1, 2 and 5, shows a higher sustainable throughput when the maximum BTL is lower even if there is a higher accumulated BTL. Comparing scenario 1, 3 and 4, it shows that accumulated BTL affects the sustainable throughput when all scenarios are having same amount of maximum BTL. Further analysis of the simulation

results show that the performance degradation (in terms of WLAN delay and number of retransmission attempts) occurs mainly in the link that carries the maximum BTL. Therefore these results show that the maximum BTL together with an accumulated BTL should be considered to measure the effect of the existing load. It further shows that the average BTL is not a good indicator for these measurements.

10.4 Effect of RTS Threshold – SF & MF Distributions

This section details the analysis of results when changing the RTS threshold. These results are compared for SP and FRDM scenarios in the basic topologies (Figure 5-1). An audio, a video and a FTP download have been used as application flows. Table 10-2 shows how RTS/CTS handshake is going to be enabled or disabled for different flows, when using different RTS thresholds. In the MF distribution, both audio and video are directed via one path and the FTP download is completed over the other path while using the FRDM paths. In case of SP, all three flows (audio, video and FTP) are forwarded over the single path.

Table 10-2 Applicability of RTS threshold for different flows

	Audio	Video	TCP-Data	TCP-ACK
Packet size @ Application, bytes	20	21 ~ 125	2264	-
Packet size @ Physical layer, bytes (UDP = 8, TCP = 20, IP = 20, MAC = 28, PHY = 24)	100	101 ~ 205	2356	92
RTS Threshold = 80 bytes (enabled - √ /disabled - ×)	√	√	√	√
RTS Threshold = 140 bytes (enabled - √ /disabled - ×)	×	× if < 140 √ (> 140)	√	×
RTS Threshold = None (enabled - √ /disabled - ×)	×	×	×	×

Table 10-3 and Table 10-4 show the evaluated parameters for both RDM and SP when using three flows simultaneously. These results show that there are higher numbers of packet losses and retransmission attempts done at the WLAN when RTS/CTS messages are disabled (i.e. RTS threshold is set to none). The hidden node problem causes the loosing of more packets due to collisions in the absence of RTS/CTS messages. This is badly affecting the FTP download due to loss of TCP-Data segments. These results conclude the following.

- *Audio Delay:* For SP, audio delay increases when RTS/CTS are enabled for all 3 flows (i.e. RTS threshold is set to 80). The lowest audio delay with the absence of RTS/CTS messages should not be the best case since it increases the number of packet lost significantly. For RDM, the lowest audio delay is shown when RTS is disabled for audio packets while RTS is enabled for part of the video packets and all TCP-Data. This avoids the collisions of larger packets due to hidden node problem created at the end nodes. In all cases, RDM performs better than SP due to the distribution of the load and thus alleviating the contention and the number of packets dropped.

- *Video Delay:* The behavior of video also shows similar characteristics like audio for both SP and RDM scenarios.
- *FTP DRT:* For SP, FTP download has been cancelled after several TCP timeouts in the absence of RTS/CTS messages. This is due to dropping TCP-Data due to collisions. Since the TCP-Data is the largest packet that takes more time to transmit, it is very likely to collide with other smaller audio and video packets that traverse more frequently in the network. In case of RDM, the TCP-Data is not transmitted together with audio and video packets. Even in RDM case, there occur packet collisions due to the hidden node problem when immediate neighbors from both paths try to access the end nodes (i.e. the node S and the node D). Therefore, FTP download is faster when all packets are transmitted with RTS/CTS enabled.

Table 10-3 MF Distribution – application performance vs RTS threshold – part I

RTS Threshold	Audio Delay, ms		Video Delay, ms		FTP DRT, sec	
	SP	RDM	SP	RDM	SP	RDM
80	154.562	123.143	48.714	20.302	157.21	91.67
140	133.474	116.741	32.600	17.401	973.28	512.55
None	131.452	119.432	28.460	19.562	X	852.00

Table 10-4 MF distribution – application performance vs RTS threshold – part II

RTS Threshold	Total num. of data packets Dropped @ WLAN		Total num. of retransmission attempts @ WLAN		Total num. of TCP timeouts		Total num. of TCP retransmissions	
	SP	RDM	SP	RDM	SP	RDM	SP	RDM
80	558	116	72,308	31,699	30	18	50	31
140	729	190	152,058	84,669	146	70	234	119
None	1,727	708	156,420	117,031	123	128	184	198

In SF distribution, individual flow is distributed among RDM paths using 1:1 distribution since both RDM paths have similar characteristics in basic topologies. When using the SP, all the packets are sent over a single path.

Table 10-5 SF distribution – application performance vs RTS threshold

RTS Threshold	Audio Delay, ms		Video Delay, ms		FTP DRT, sec	
	SP	RDM	SP	RDM	SP	RDM
80	117.928	116.415	15.325	15.644	85.93	39.17
140	112.546	111.316	13.439	12.281	85.93	39.17
None	112.507	111.316	13.430	11.703	64.66	638.18

Table 10-5 concludes following for the SF distribution with the change of the RTS threshold.

- *Audio Delay & Video Delay:* For both SP and RDM, audio and video delays increase when RTS/CTS are enabled due to the fact that it takes longer time to exchange RTS/CTS messages for each transmission. When all video packets are transmitted with RTS/CTS enabled, RDM does not show performance improvement compared to the SP. The video packets are transmitted

continuously at the rate of 10 packets per second. Therefore, both the node S and the node D receive video packets simultaneously from both paths and it takes longer time to do RTS/CTS handshake. This causes higher delay for video packets. This is further justified with larger number of WLAN retransmission attempts used by the end nodes. In contrast to video transmission, the bulk of audio packets are transmitted at once and then there is no transmission during the silence period. When using RDM paths, splitting helps to alleviate the contention of audio transmission. Therefore, audio shows a performance improvement compared to the SP in all cases.

- *FTP DRT:* For SP, FTP download is faster with the absence of RTS/CTS messages. In contrast, FTP DRT increases drastically for RDM scenario with splitting. Further investigation shows that the most of the TCP-Data are lost when receiving at the node S. This happens when the node 1 and the node 7 transmit TCP-Data at the same time to the node S and the both packets are lost due to collisions. The simultaneous transmissions and the probability of collisions increase due to having same path characteristics and the TCP-Data of 2356 bytes takes longer time to transmit in 1 Mbps PHY mode. The longer transmission time causes the packets collisions even after the exponential back-off used by the node 1 and the node 7. This behavior is explained in detail in section 10.2.1. The above two problems can be avoided by using a bulk distribution rate or a higher PHY mode. FTP DRT with RDM paths has been reduced to 79.12 seconds from 638.18 seconds when using bulk distribution of 10:10 with the absence of RTS/CTS messages. The bulk distribution reduces the simultaneous transmission of TCP-Data to the node S from both paths.

As discussed, transmission of larger packets (e.g. TCP-Data in the FTP download) together with smaller packets without enabling RTS/CTS deteriorates the TCP transmission very badly, especially for the SP. Further, the enabling RTS/CTS do not affect the performance badly even for the SF distribution.

10.5 Overhead Comparison of Promiscuous vs Non-promiscuous

As shown in Figure 10-8, 4 laptops are setup closer to each other. Each laptop uses 802.11b wireless cards with the *orinocao* chip set. Laptop A starts sending ping packets to laptop B while laptop C starts sending ping packets to laptop D. All four laptops are within each other's communication range. The following results are taken in laptop A.

- *Total CPU usage (%):* This statistic is collected by using *TOP* command in Linux. It shows the CPU usage of all running applications as a percentage at each 2 seconds. The results are collected during a 60 second period.
- *Average RTT (ms):* The output of ping command shows the average round trip time of each packet that it sends. This is also measured by collecting 60 ping packets.

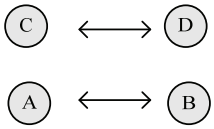


Figure 10-8 4 Test setup: All 4 laptops are within each other's transmission range

The communication between laptop A and B is interfered by the communication between laptop C and D and vice versa. The above results are repeated by setting the wireless card of laptop A to promiscuous mode enabled (promiscuous) and disabled (non-promiscuous). For each run, the following 3 different cases are repeated by changing the sending rate of ping packets.

- Case 1: sending one ping packet per second by both node A and C
- Case 2: sending 100 ping packets per second by node A and 1 ping packet per second by node C
- Case 3: sending 100 ping packets per second by both node A and C

Table 10-6 CPU usage and RTT variation (promiscuous mode disabled and enabled)

	Total CPU usage (%)		Average RTT (ms)	
	Non-promiscuous	Promiscuous	Non-promiscuous	Promiscuous
Case 1	4.71 ±0.47	4.77 ±0.56	1.89 ±0.20	1.90 ±0.29
Case 2	78.20 ±.85	78.28 ±1.18	1.74 ±0.12	1.77 ±0.21
Case 3	77.64 ±1.2981	77.14 ±3.3032	2.17 ±0.15	2.21 ±0.28

Table 10-6 shows the average of 10 runs. It shows a higher variation of the CPU usage and the average RTT when the wireless card of laptop A is set to promiscuous mode.

When a wireless card is set to promiscuous mode, the chipset of the card (MAC layer) has to send all the packets that are not destined to the node itself also to the network layer. In the above setup, laptop A receives all the packets sent by laptop B together with packets sent by laptop C and laptop D. When the wireless card of laptop A works on the promiscuous mode, the MAC layer has to forward all the packets that it receives to the network layer. Then the network layer has to process these packets in order to filter the packets destined to the node itself. In case of non-promiscuous mode, the filtering of packets is done at the chipset itself. Therefore a higher variation of CPU usage observed with promiscuous mode could be justified with the following reasons:

- The filtering done in the chipset might be faster than the filtering done at the network layer.
- Use of the promiscuous mode can result in taking more processing time to send unwanted packets form the MAC layer to the network layer.

11. List of Figures

12. List of Tables

13. Glossary

A		
	ACK	Acknowledgment
	AODV	Ad hoc On-demand Distance Vector Routing
	AODV-BR	Backup Routing for AODV
	AODVM	AODV Multipath routing
	AOMDV	Ad hoc On-demand Multipath Distance Vector routing
	APR	Alternate Path routing
	AR	Access Router
	A-TCP	Ad hoc TCP
B		
	BTL	Background Traffic Load
	BSPP	Brigade des Sapeurs-Pompiers de Paris
C		
	CA	Congestion Avoidance
	CBR	Continuous Bit Rate
	CWND	Congestion Window of TCP
	CS	Carrier Sensing
	CSMA/CA	Carrier Sense Multiple Access/Collisions Avoidance
D		
	D	Destination
	DCF	Distributed Coordinated Function
	DRT	Download Response Time
	DSCP	Differentiated Services Code Point
	DSR	Dynamic Source Routing
	DYMO	Dynamic Manet On-demand protocol
	DYMOM	Dynamic Manet On-demand Multipath protocol
E		
	ECN	Explicit Congestion Notification
	ELFN	Explicit Link Failure Notification
	ENIC	ENhanced Inter-layer Communication and control
	ESPAR	Electronically Steerable Passive Array Radiator
F		
	FID	Flow ID
	FRDM	Full Radio Disjoint Multipath
	FTP	File Transfer Protocol
G		
	GPS	Global Positioning System
H		
	HBTL	High Background Traffic Load
	HTTP	Hyper Text Transfer Protocol
I		
	I	Intermediate node
	ICI	Interface Control Information
	ICMP	Internet Control Message Protocol
	IEEE	Institute of Electrical & Electronics Engineers

	IETF	Internet Engineering Task Force
	IMA	Internal Model Access
	INL	Interfering Neighbor List
	IP	Internet Protocol
	IS	Independent Set
L		
	LAR	Location Aided Routing
	LBTL	Low Background Traffic Load
	LP	Linear Programme
	LSR	Link State Routing
M		
	MAC	Medium Access Control
	MACA	Multiple Access Collision Avoidance
	MACAW	Multiple Access Collision Avoidance for WLAN
	MANET	Mobile Ad hoc NETwork
	MF	Multiple Flow
	MP	Multipath
	MP-DSR	Multipath Dynamic Source Routing
	MP-ODP	Multipath Routing for On-Demand Protocols
	MPTCP	MultiPath TCP
N		
	NI	Node Interference
	NL	Node Load
	NP	Non Polynomial
	NRDM	Non Radio Disjoint Multipath
	NS-2	Network Simulator - 2
O		
	OLSR	Optimized Link State Routing
	OMR	On-Demand Multipath Routing for mobile ad hoc networks
	OrgBW	Original bandwidth
	ORT	Object Response Time
P		
	PDV	Packet Delay Variation
	PE-C	Path Evaluation Confirmation
	PE-RERR	Path Evaluation RERR
	PE-RREP	Path Evaluation RREP
	PE-RREQ	Path Evaluation RREQ
	PE-RT	Path Evaluation Routing Table
	Pid	Path identification number
	PHY	PHYsical layer
	PRDM	Partial Radio Disjoint Multipath
	PRT	Page Response Time
	PL	Path Load
Q		
	QoS	Quality of Service
R		
	RAM	Random Access Memory
	RCM	Route Confirmation Message
	RDER	Route Discovery Error
	RDM	Radio Disjoint Multipath
	RDT	Route Discovery Time
	RE	Route Element
	REB	Route Element Block
	RemBW	Remaining Bandwidth

	RERR	Route Error
	RFC	Request For Comments
	RFN	Route Failure Notification
	RREP	Route Reply
	RREQ	Route Request
	RTO	Retransmission Time Out
	RTT	Round Trip Time
	RTS/CTS	Request To Send/Clear To Send
	RWM	Random Waypoint Model
S		
	S	Source
	SDP	Shortest Delay Path
	STDMA	Spatial Time Division Multiple Access
	SF	Single Flow
	SMR	SMR: Split Multipath Routing
	SNR	Signal to Noise Ratio
	SS	Slow Start Phase
	SS-threshold	Slow Start threshold
	SP	Single Path
	SUM	Simultaneous Use of Multipath routing
T		
	TBRPF	Topology Based Reverse Path Forwarding
	TCP	Transmission Control Protocol
	TCP-ACK	TCP ACKnowledgement
	TCP-Data	TCP Data packet
	TCP-Feedback	TCP-F
	TDMA	Time Division Multiple Access
	TETRA	TErrestrial TRunked Radio communication standards
	ToS	Type of Service
	TCP/IP	Transmission Control Protocol/Internet Protocol
U		
	UDP	Universal Datagram Protocol
V		
	VoIP	Voice over IP
W		
	WDS	Wireless Distribution System
	WLAN	Wireless Local Area Networks
	WLAN-ACK	WLAN ACKnowledgement
	WMN	Wireless Mesh Network
	WSN	Wireless Sensor Network
	WWW	World Wide Web browsing
Z		
	ZDR	Zone Disjoint Routes

13.1.1 List of Symbols

C	Connectivity Graph
Cap_{ij}	Link capacity of the link l_{ij}
d_{ij}	Distance between node n_i and node n_j
d	Distance between the transmitter and the receiver in Km
f_{ij}	Amount of directional flow on link l_{ij}
f	Frequency in GHz
G	A graph
$HC_{current}$	Number of hops to the source in the current RREQ message
$HC_{existing}$	Number of hops to the source in the routing table
I_r	Interference index of the r^{th} path
I_{1r}	Mutual interference of the r^{th} path w.r.t. the primary path
INL_i	Interfering neighbour list of the i^{th} node
IS_m	m^{th} independent set
K	Total number of independent sets in a given network
L_C	Wireless links in the connectivity graph C
l_{ij}	Link between n_i and n_j
N	Number of nodes in a wireless network
N_a	Ambient noise
NL_{acu}	Accumulated Node Load (NL)
$NL_{\max}$	Maximum NL
NL_{ir}	NL of the i^{th} node in the r^{th} path
$(NL_{\max})_{RREQ}$	Maximum NL in the RREQ message
$(NL)_{i\,node}$	Maximum NL of the i^{th} node
$(NL_{acu})_{RREQ}$	Accumulated NL in the RREQ message
n_i	i^{th} node
n_d	Destination node
N_j	Total noise at node n_j
$N_{l_{ij}}$	Number of links that are interfering with the link l_{ij}
n_s	Source node
L_r	All possible links existing in the r^{th} path
P_{Loss}	Path loss in dB
P_r	Primary path
PL_r	Path load of the r^{th} path
P_{Tx}	Transmitting power at the sender

R_i	Communication range of node n_i
R'_i	Interference range of node n_i
R_{cs}	Carrier Sensing Range
Sen_{Rx}	Receiver sensitivity
SNR_{ij}	Signal-to-noise ratio at the node n_j, when node n_i is transmitting
SNR_{thresh}	Signal-to-noise ratio threshold
SS_{ij}	Received signal strength at node n_j due to node n_i's transmission
T_{ir}	Traffic load of i^{th} link in the r^{th} path
T_r	Traffic load of r^{th} path
α	Weighing factor to weight the interference of a path
β	Weighing factor to weight the BTL of a path
λ_n	Fraction of time that can be used by the independent set IS_n
ω_{ij}^{pq} / Ω_{ij}^{pq}	Weight factor showing the interference from the link l_{pq} to the link l_{ij}
V	The vertex-set of a graph G

14. References

1. Perkins, C. E., *Ad Hoc Networking*. 1st ed. 2000, NJ: Addison-Wesely.
2. Timm-Giel, A., Kuladinithi, K., Hofmann, P., and Goerg, C., *Wireless and Ad Hoc Communications Supporting the Firefighter*, in *16th IST Mobile Summit*. June, 2006: Myconos, Greece [DVD p.1-5].
3. Kuladinithi, K., Timm-Giel, A., and Görg, C., *Potential Productivity Gains in the Field through Wireless Technologies*, in *10th International Conference on Concurrent Engineering: Research and Applications (CE)*. July 2003: Madeira, Portugal. p. 725-731.
4. Kuladinithi, K., Timm-Giel, A., and Görg, C., *Mobile Ad Hoc Communication in AEC Industry.* IT-Con Journal on Mobile Computing in Construction, August 2004. **vol. 9**: p. 313-323.
5. Kuladinithi, K. and Görg, C. *Tutorial on Mobile Ad Hoc Networks*. in *1st Regional Conference on ICT and E-paradigms*. June 2004. Colombo, Sri Lanka.
6. Mueller, S., Tsang, R.P., and Ghosal, D., *Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges Performance Tools and Applications to Networked Systems.* Springer Berlin / Heidelberg, 2004.
7. Kuladinithi, K., Udugama, A., and Görg, C., *On Demand Self Organising Ad Hoc Networks - Implementation Architectures*, in *12th WWRF meeting 2004*. Nov 2004: Toronto, Canada.
8. Mushtaq, Q., An, C., Kuladinithi, K., and Görg, C. *QoS Aware Routing for Wireless Ad Hoc Networks*. in *Baltic Conference on Advanced Topics in Telecommunication*. Aug. 2008. Tartu, Estonia. **vol. 1**: p. 19-31.
9. IETF-Working-Group, *MANET: Mobile Ad hoc NETworks*, available at *www.ietf.org/html.charters/manet-charter.html,* accessed on July 2008.
10. Clausen, T. and Jacquet, P., *Optimized Link State Routing Protocol (OLSR), Request For Comments (Proposed Standard) 3626*. Oct. 2003, Internet Engineering Task Force.
11. Ogier, R., Templin, F., and Lewis, M., *Topology Dissemination Based on Reverse-Path Forwarding, Request For Comments (Proposed Standard) 3684*. Feb. 2004, Internet Engineering Task Force.
12. Perkins, C., Belding-Royer, E., and Das, S., *Ad hoc On-demand Distance Vector routing, Request For Comments (Proposed Standard) 3561*. July 2003, Internet Engineering Task Force.
13. Johnson, D. B., Hu, Y., and Maltz, D. A., *The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks for IPv4, Request For Comments (Proposed Standard) 4728*. Feb 2007, Internet Engineering Task Force.
14. Chakeres, I. and Perkins, C., *Dynamic MANET On-demand (DYMO) Routing*. Mar. 2009, IETF Draft draft-ietf-manet-dymo-17, available at

http://www.ietf.org/internet-drafts/draft-ietf-manet-dymo-17.txt, accessed on July 2009.
15. Kuladinithi, K., Fikouras, N.A., and Görg, C., *Filters for Mobile Ad hoc Networks (NOMADHOC)*. 2003, IETF Drfat, draft-nomadhoc-manet-filters-00.txt.
16. Kuladinithi, K., Becker, M., Görg, C., and Das, S. *Radio Disjoint Multi-Path Routing in MANET*. in *CEWIT (Center of Excellence in Wireless and Information Technology) 2005 Conference*. Dec 2005. Stony Brook.
17. Das, S. R., Perkins, C. E., Royer, E. M., and Marina, M. K., *Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks.* IEEE Personal Communications Magazine special issue on Ad hoc Networking, 2001: p. 16-28.
18. Lee, S.J., Royer, E. M., and Perkins, C. E., *Scalability Study of the Ad Hoc on-Demand Distance Vector Routing.* ACM/Wiley International Journal of Network Management, 2003: p. 97-114.
19. Mushtaq, Q., *QoS Aware Routing in Wireless Ad-hoc Networks*, Master Thesis, ComNets, University of Bremen, Germany, Aug. 2008.
20. Ng, P. and Liew, S., *Throughput Analysis of IEEE802.11 Multi-hop Ad Hoc Networks.* IEEE/ACM Trans. Networking, June 2007. **vol. 15**: p. 309-322.
21. Hofmann, P., An, C., Loyola, L., and Aad, I. *Analysis of UDP, TCP and Voice Performance in IEEE 802.11b Multihop Networks*. in *European Wireless 2007*. March 2007. Paris, France [DVD p.1-5].
22. Chakeres, I. D. and Belding-Royer, E. M., *Transparent Influence of Path Selection in Heterogeneous Ad Hoc Networks*, in *15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)* Sep 2004: Barcelona, Spain. p. 885-889.
23. Leung, R., Liu, J., Poon, E., Chan, A. L. C., and Li, B., *MP-DSR: a QoS-aware Multi-path Dynamic Source Routing Protocol for Wireless Ad-hoc Networks*, in *26th Annual IEEE Conference on Local Computer Networks*. 2001: Florida, USA. p. 132-141.
24. Chakeres, I. and Perkins, C., *Dynamic MANET On-demand (DYMO) Routing*. July 2007, IETF Draft draft-ietf-manet-dymo-10, available at http://tools.ietf.org/html/draft-ietf-manet-dymo-10, accessed on May 2009.
25. Pearlman, M. R. and Haas, Z. J., *Improving the Performance of Query-based Routing Protocols Through Diversity Injection*, in *proceedings of IEEEWireless Communications and Networking Conference (WCNC'99)*. Sep 1999: New Orleans, USA. **vol. 3:** p. 1548-1552.
26. Marina, M. K. and Das, S. R., *On-demand Multipath Distance Vector Routing for Ad Hoc Networks*, in *Proceedings of the International Conference for Network Protocols (ICNP)*. Nov. 2001: Riverside, CA. p. 14-23.
27. Lee, S.J. and Gerla, M., *Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks*, in *Proceedings of IEEE ICC 2001*. 2001: Helsinki, Finland. p. 3201-3205.
28. Soliman, H., Tsirtsis, G., Montavont, N., Giaretta, G., and Kuladinithi, K., *Flow Bindings in Mobile IPv6 and Nemo Basic Support*. July 2009, IETF Draft, draft-ietf-mext-flow-binding-03.txt.
29. Ye, Z., Krishnamurthy, S. V., and Tripathi, S.K., *Effects of Multipath Routing on TCP Performance in Ad Hoc Networks.* IEEE Global Telecommunications Conference (GLOBECOM), 2004. **vol. 6**: p. 4125-4131.

30. Wang, L., Shu, Y., Dong, M., Zhang, L., and Yang, O.W.W., *Adaptive Multipath Source Routing in Ad Hoc Networks*, in *proceedings of IEEE International Conference on Communications*. 2001: Helsinki, Finland. p. 857-861.
31. Batassi, S., *Flow Distribution Algorithms for Multipath Routing*, Mini Project Report, ComNets, University of Bremen, Germany, Sep 2006.
32. Sun, X., *Simulation and Analysis of Packets Replication over Multi-Path Routing in MANET*, Master Thesis, ComNets, University of Bremen, Germany, Jan 2007.
33. Lee, S.J. and Gerla, M., *AODV-BR: Backup Routing in Ad Hoc Networks*, in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*. Sep. 2000: Chicago, USA. **vol. 3:** p. 1311-1316.
34. Marina, M. K. and Das, S. R., *Ad Hoc On-demand Multipath Distance Vector Routing*. 2003, Technical report, Computer Science Department, Stony Brook University.
35. Ye, Z., Krishnamurthy, S. V., and Tripathi, S. K., *A Framework for Reliable Routing in Mobile Ad Hoc Networks*, in *IEEE INFOCOM*. March 2003: Sanfrancisco CA. p. 270-280.
36. Nasipuri, A., Castaneda, R., and Das, S. R., *Performance of Multipath Routing for On-demand Protocols in Mobile Ad Hoc Networks.* ACM/Kluwer Mobile Networks and Applications (MONET) Journal, 2001. **vol. 6**(4): p. 339-349.
37. Wu, K. and Harms, J., *On-Demand Multipath Routing for Mobile Ad Hoc Networks*, in *Proceedings of 4th European Personal Mobile Communication Conference (EPMCC)* Feb 2001: Vienna, Austria. p. 1-7.
38. Koltsidas, G., Pavlidou, F.N., Kuladinithi, K., Timm-Giel, A., and Goerg, C., *Investigating the Performance of a Multipath DYMO Protocol for Ad-hoc Networks*, in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. Sep 2007: Athens, Greece. p. [DVD p.1-5].
39. Saha, D., Toy, S., Bandyopadhyay, S., Ueda, T., and Tanaka, S., *An Adaptive Framework for Multipath Routing via Maximally Zone-disjoint Shortest Path in Ad Hoc Wireless Networks with Directional Antenna*, in *IEEE Global Telecommunications Conference (GLOBECOM)*. 2003. p. 226-230.
40. Ueda, T., Tanaka, S., Saha, D., Roy, S., and Bandyopadhyay, S. A., *A Rotational Sector-based, Receiver-oriented Mechanism for Location Tracking and Medium Access Control in Ad Hoc Networks using Directional Antenna*, in *Proceedings of the IFIP conference on personal wireless communications PWC 2003*. Sep 2003: Venice, ITALY. p. 601-610.
41. Park, V. D. and Corson, M. S., *A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks*, in *In Proceedings of IEEE Infocom, 1997*: Kobe. p. 1405-1413.
42. Raju, J. and Garcia-Luna-Aceves, J. J., *A New Approach to On-demand Loop-free Multipath Routing*, in *Proceedings of Computer Communications and Networks, 1999*: Boston, MA. p. 522-527.

43. Ko, Y. B. and Vaidya, N. H., *Location-aided Routing (LAR) in Mobile Ad Hoc Networks.* Wireless Networks, 2000. **vol. 6**(4): p. 307-321.
44. Draves, R., Padhye, J., and Zill, B., *Routing in Multi-radio, Multi-hop Wireless Mesh Networks*, in *International Conference on Mobile Computing and Networking, 2004*: Philadelphia, PA, USA. p. 114-128.
45. Raniwala, A., Gopalan, K., and Chiueh, T., *Centralized Channel Assignment and Routing Algorithms for Multi-channel Wireless Mesh Networks.* SIGMOBILE Mobile Computer Communication Review, 2004. **vol. 8**: p. 50-65.
46. Kyasanur, P., Jungmin, S., Chereddi, C., and Vaidya, N. H., *Multichannel Mesh Networks: Challenges and Protocols.* IEEE Personal Communications, 2006. **vol. 13**: p. 30-36.
47. Subramanian, A. P., Gupta, H., Das, S. R., and Cao, J., *Minimum Interference Channel Assignment in Multi-Radio Wireless Mesh Networks.* IEEE Transactions on Mobile Computing (TMC), June 2007: p. 481-490.
48. Gast, M.S., *802.11 Wireless Networks: The Definitive Guide*. 2nd ed. April 2005: O'Reilly Media, Inc. pp 298-310.
49. Pearlman, M. R., Haas, Z. J., Sholander, P., and Tabrizi, S. S., *On the Impact of Alternate Path Routing for Load Balancing in Mobile Ad Hoc Networks*, in *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing* 2000: Boston, Massachusetts. p. 3-10.
50. Kuladinithi, K. and Görg, C. *DYMO Implementation on OPNET - Analysis of Results*. in *63rd IETF Meeting* Aug 2005. Paris, France.
51. *The OPNET Simulator*, available at www.opnet.com accessed on Oct 2008.
52. Kuladinithi, K., Timm-Giel, A., and Görg, C. *Implementation and Analysis of Radio Disjoint Multi-path Routing over DYMO in OPNET*. in *OPNETWork 2007*. Aug 2007. USA.
53. Clausen, T., Dearlove, C., and Adamson, B., *Jitter Considerations in Mobile Ad Hoc Networks (MANETs), Request For Comments (Proposed Standard) 5148*. Feb. 2008, Internet Engineering Task Force.
54. Marsh, M., *Policy Routing Using Linux (Professional)*.
55. Kuladinithi, K., Fikouras, N. A., Könsgen, A., Timm-Giel, A., and Görg, C., *Enhanced Terminal Mobility through the Use of Filters for Mobile IP*, in *Proceedings of the Summit on Mobile and Wireless Communications (IST Summit)*. June 2003: Aveiro, Portugal. **vol. 2:** p. 53-57.
56. Fikouras, N. A., Udugama, A., Görg, C., Zirwas, W., and Eichinger, J. M. *Experimental Evaluation of Load Balancing for Mobile Internet Real-Time Communications*. in *Proceedings of the Sixth International Symposium on Wireless Personal Multimedia Communications (WPMC)*. Oct 2003. Yokosuka, Kanagawa, Japan. **vol. 3:** p. 113-118.
57. Stevens, W. R., *TCP/IP Illustrated, Volume 1*. 1994: Addison-Wesley.
58. Stevens, W. R., *TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms, Request For Comments (Proposed Standard) 2001*. Jan. 1997, Internet Engineering Task Force.
59. Holland, G. and Vaidya, N., *Analysis of TCP Performance over Mobile Ad Hoc Networks*, in *5th annual ACM/IEEE International Conference on Mobile Computing and Networking*. August 1999. p. 219-230.
60. Xu, S. and Saadawi, T., *Does IEEE 802.11 MAC Protocol Work Well in Multi-hop Wireless Networks?* IEEE Communications, June 2001. **vol. 39**(6): p. 130-137.

61. Gerla, M., Tang, K., and Bagrodia, R., *TCP Performance in Wireless Multi-hop Networks*, in *IEEE Workshop on Mobile Computer Systems and Applications*. Feb 1999. p. 41-50.
62. Jiang, R., Gupta, V., and Ravishankar, C. V., *Interactions Between TCP and the IEEE 802.11 MAC Protocol.* Proceedings of the DARPA Information Survivability Conference and Exposition (DISCEX´03), 2003. **vol. 1**: p. 273-282.
63. Chandran, K., Raghunathan, S., Venkatesan, S., and Prakash, R., *A Feedback-based Scheme for Improving TCP Performance in Ad Hoc Wireless Neworks.* IEEE Personal Communications Magazine, Feb 2001. **vol. 8**(1): p. 34-39.
64. Dyer, T. D. and Boppana, R. V., *A Comparison of TCP Performance over Three Routing Protocols for Mobile Ad Hoc Networks*, in *Proceedings of ACM MobiHoc*. Oct, 2001. p. 56-66.
65. Sun, D. and Man, H., *ENIC – An Improved Reliable Transport Scheme for Mobile Ad Hoc Networks.* IEEE Global Telecommunications Conference, Nov. 2001. **vol. 5**: p. 2852–2856.
66. Liu, J. and Singh, S., *ATCP: TCP for Mobile Ad Hoc Networks.* IEEE Journal on Selected Areas in Communications, July 2001. **vol. 19**(7): p. 1300-1315.
67. Wang, Y. and Garcia-Luna-Aceves, J.J., *Throughput and Fairness in a Hybrid Channel Access Scheme for Ad Hoc Networks.* Proceedings of the IEEE Wireless Communication and Networking Conference (WCNC), 2003. **vol. 2**: p. 988-993.
68. Luo, H., Medvedev, P., Cheng, J., and Lu, S., *A Self-Coordinating Approach to Distributed Fair Queueing in Ad Hoc Wireless Networks.* IEEE INFOCOM, 2001. **vol. 3**: p. 1370-1379.
69. Kanodia, V., Li, C., Sabharwal, A., Sadeghi, B., and Knightly, E. *Distributed Multi-hop Scheduling with Delay and Throughput Constraints*. in *proceedings of ACM MOBICOM*. 2001.
70. Pearlman, M.R., Haas, Z.J., Sholander, P., and Tabrizi, S.S. *Alternate Path Routing in Mobile Ad Hoc Networks*. in *Proceedings of IEEE MILCOM*. Oct. 2000. LA, USA.
71. Lim, H., Xu, K., and Gerla, M., *TCP Performance over Multipath Routing in Mobile Ad Hoc Networks.* IEEE ICC, 2003. **vol. 2**: p. 1064–1068.
72. IETF-Working-Group, *MPTCP : Multipath TCP*, available at http://www.ietf.org/dyn/wg/charter/mptcp-charter.html, accessed on Nov 2009.
73. Bagnulo, M., Burness, L., Eardley, P., García-Martínez, A., Valera, F., et al., *Joint Multi-path Routing and Accountable Congestion Control*, in *ICT Mobile Summit* June 2009: Santander, Spain [DVD p.1-5].
74. Postel, J., *User Datagram Protocol, Request For Comments (Proposed Standard) 768*. Aug. 1980, Internet Engineering Task Force.
75. Kuladinithi, K., Udugama, A., Fikouras, N. A., Timm-Giel, A., and Görg, C., *Experimental Evaluation of AODV Implementations*, in *The Center of Excellence in Wireless & Information Technology (CEWIT)*. Oct 2004: New York.

76. Gupta, V., Krishnamurthy, S.V., and Faloutsos, M. *Improving the Performance of TCP in the Presence of Interacting UDP Flows in Ad Hoc Networks*. in *IFIP Networking 2004*. 2004. Athens, Greece.
77. 3GPP2, *CDMA 2000 Evaluation Methodology (3GPP2 documentation C.R1002-0)*. Dec, 2004.
78. Baase, S. and Gelder, A. V., *Bin Packing*, in *Computer Algorithms - Introduction to Design & Analysis*. 2000, Addison-Wesley. p. 572-577.
79. Skiena, S. S., *Bin Packing*, in *The Algorithm Design*. 1997, Springer-Verlag New York. p. 374-376.
80. Wenning, B., *Context-based Routing in Dynamic Networks*. Aug. 2009, PhD Thesis, University of Bremen.
81. TETRA, *Terrestrial Trunked Radio*. 2007, available at, http://www.tetra-association.com/, accessed on July 2009.
82. Appelbaum, A., *Clearer Signals Through the Smoke*, IEEE Spectrum, Feb 2005 available at http://www.spectrum.ieee.org/feb05/2943, accessed on Sep 2007.
83. *wearIT@work project*, available at http://www.wearitatwork.com, accessed on May 2009.
84. Timm-Giel, A., Kuladinithi, K., and Görg, C., *WearIT@work: Communications for the Mobile Worker Equipped with Wearable Computing*, in *15th IST Mobile Summit*. June 2005 Dresden, Germany [DVD p.1-5].
85. Timm-Giel, A., Kuladinithi, K., Hofmann, P., Bettstetter, C., Ok, R., et al., *Research in Wireless Communications for Wearable Computing*, in *Forum on Applied Wearable Computing (IFAWC)*. Mar, 2005: Zürich, Switzerland. p.187-188.
86. Hofmann, P., Bettstetter, C., Kuladinithi, K., Timm-Giel, A., Görg, C., et al., *Requirements of Wireless Communication for Wearable Computing at Work*, in *Forum on Applied Wearable Computing (IFAWC)*. Mar 2005: Zürich, Switzerland. p. 179-180.
87. Hofmann, P., Kuladinithi, K., Timm-Giel, A., and Görg, C., *Performance of IEEE 802.11 Wireless Technologies in Airplane Maintenance*, in *Forum on Applied Wearable Computing (IFAWC)*. Mar 2006: Bremen, Germany. p. 125–130.
88. Timm-Giel, A., Kuladinithi, K., and Görg, C. *Self-configuring Communication Service Module for Wearable Computers*. in *WWRF15 Meeting*. 2005. Paris.
89. Hofmann, P., Kuladinithi, K., Timm-Giel, A., Görg, C., Bettstetter, C., et al., *Are IEEE 802 Wireless Technologies Suited for Fire Fighters?*, in *12th European Wireless Conference*. April, 2006: Athens, Greece. [DVD p1-5].
90. Jacobson, V., *Modified TCP Congestion Avoidance Algorithm, end2end-interest mailing list, available at ftp://ftp.isi.edu/end2end/end2end-interest-1990.mail*. April 30, 1990.
91. Ludwig, R. and Meyer, M., *The Eifel Detection Algorithm for TCP, Request For Comments (Proposed Standard) 3522*. April 2003, Internet Engineering Task Force.
92. Ludwig, R. and Gurtov, A., *The Eifel Response Algorithm for TCP, Request For Comments (Proposed Standard) 4015*. Feb. 2005, Internet Engineering Task Force.
93. Floyd, S., *An Extension to the Selective Acknowledgement (SACK) Option for TCP, Request For Comments (Proposed Standards) 2883*. July 2001, Internet Engineering Task Force.

94. Gupta, P. and Kumar, P.R., *The Capacity of Wireless Networks.* IEEE Transactions on Information Theory, 2000. **vol. 46**: p. 388-404.
95. Li, J., Blake, C., Couto, D. S. J. De, Lee, H. I., and Morris, R., *Capacity of Ad Hoc Wireless Networks*, in *ACM SIGMOBILE*. July 2001: Rome, Italy. p. 61-69.
96. Xu, K., Gerla, M., and Bae, S., *How Effective is the IEEE 802.11 RTS/CTS Handshake in Ad Hoc Networks?* IEEE GLOBECOM, November 2002. **vol. 1**: p. 17-21.
97. Williams, H. P., *Model Building in Mathematic Programming*. 4th ed. Nov 2006: John Wiley & Sons Ltd. pp 59-92.
98. Jain, K., Padhye, J., Padmanabhan, V., and Qiu, L., *Impact of Interference on Multi-hop Wireless Network Performance*, in *Mobicom 2003*. Sep 2003: San Diego, California, USA. p. 66-80.
99. Coenen, T., Graaf, M. de, and Boucherie, R. J., *An Upper Bound on Multi-hop Wireless Network Performance*, in *20th International Teletraffic Congress, ITC 20*. June 2007: Ottawa, Canada. p. 335-347.
100. Haan, R. D., Boucherie, R. J., and Ommeren, J. V., *The Impact of Interference on Optimal Multi-path Routing in Ad Hoc Networks*, in *20th International Teletraffic Congress, ITC 20*. June 2007: Ottawa, Canada. p. 803-815.
101. Gibbons, A. M., *Algorithmic Graph Theory*. 1985: Cambridge University Press. pp 1-119.
102. *Link Planning for Wireless LAN (WLAN)* available at http://huizen.deds.nl/~pa0hoo/helix_wifi/linkbudgetcalc/wlan_budgetcalc.html, accessed on Mar 2008.
103. 802.11-1997, IEEE Std, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*. 1997.
104. Kuladinithi, K., An, C., Timm-Giel, A., and Görg, C. *Performance Evaluation of Radio Disjoint Multipath Routing*. in *Proceedings of the 5th Polish-German Teletraffic Symposium*. Oct. 2008. Berlin, Germany. p 171-184.
105. Kuladinithi, K., An, C., Timm-Giel, A., and Görg, C., *Performance Evaluation of Radio Disjoint Multipath Routing.* European Transactions on Telecommunications, Special Issue on the 5th Polish-German Teletraffic Symposium (PGTS 2008), Oct, 2009. **vol. 20**(7): p. 668 - 678.
106. Bianchi, G., *Performance Analysis of the IEEE 802.11 Distributed Coordination Function.* IEEE Journal on Selected Areas in Communications, 2000. **vol. 8**(3): p. 535-547.
107. Kashyap, A., Ganguly, S., and Das, S. R., *A Measurement-Based Approach to Modeling Link Capacity in 802.11-Based Wireless Networks*, in *Proc. of 13th ACM Annual International Conference on Mobile Computing and Networking (MobiCom)*. 2007: Montreal. p. [DVD p.1-12].