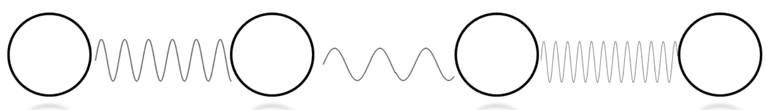
Georg-August-Universität Göttingen

Fakultät für Mathematik und Informatik

Route Record Distance Vector Protocol for Minimization of Intra-Flow Interference



Dissertation
zur Erlangung des
mathematisch-naturwissenschaftlichen Doktorgrades
’“Doctor rerum naturalium”’
der Georg-August-Universität Göttingen
im Promotionsprogramm Ph.D-Programme in Computer Science (PCS)
der Georg-August Univresity School of Science (GAUSS)

Vorgelegt von
Roman Seibel
aus
Burnooktjabrskoje (Kasachstan)

Göttingen
2013

Thesis Commitee

Prof. Dr. Dieter Hogrefe
Institute for Computer Science, Georg-August-University Göttingen

Prof. Dr. Xiaoming Fu
Institute for Computer Science, Georg-August-University Göttingen

Members of the Examination Board

Prof. Dr. Dieter Hogrefe (First Reviewer)
Institute for Computer Science, Georg-August-University Göttingen

Prof. Dr. Adreas Kassler (Second Reviewer)
Institute for Computer Science, Karlstat-University, Sweden

Prof. Dr. Xiaoming Fu
Institute for Computer Science, Georg-August-University Göttingen

Prof. Dr. Stephan Waack
Institute for Computer Science, Georg-August-University Göttingen

Prof. Dr. Carsten Damm
Institute for Computer Science, Georg-August-University Göttingen

Jun.-Prof. Dr. Konrad Rieck
Institute for Computer Science, Georg-August-University Göttingen

Date of the oral examination: 24th October 2013

Route Record Distance Vector Protocol for Minimization of Intra-Flow Interference

Abstract: The performance of wireless multi-hop networks is negatively affected by interference between neighbouring links on the same channel. Especially intra-flow interference degrades network performance to a high degree. To overcome this problem this thesis presents a routing protocol for wireless multi-hop networks with multiple radio interfaces, which minimizes intra-flow interference with significant performance improvement. We show that our route record based distance vector protocol can exploit channel diversity better than classic distance vector protocols. This protocol reduces the issue of intra-flow interference to a channel combination problem by collecting route records along a path. The challenge of channel selection belongs to classical graph labelling, specifically sequence labelling, which we solve efficiently with a Viterbi and a greedy algorithm. A theoretical and numerical analysis of the problem and solution is given complemented by empirical evidence from extensive simulation. This thesis reports substantial improvement in network performance measured in throughput and packet delivery fraction without additional routing load.

Keywords: Multi-Radio, Multi-Hop, Distance Vector, Interference, Metric, Viterbi,

Greedy

Acknowledgments

I would like to thank the following people for their support and contribution to this work:

Prof. Dr. Dieter Hogrefe and Prof. Dr. Xiaoming Fu for giving me the chance to elaborate an interesting and complex problem and contribute to the scientific field with valuable advice and guidance. Prof. Dr. Andreas Kassler for his expert perspective on this work with very valuable advice. Jun.-Prof. Dr. Konrad Rieck for his input on how scientific work should be conducted and presented.

Somayeh Taheri for her excellent contribution to this work, the countless discussions and problem solving sessions and also her husband Mostafa for his mathematical advice. Keyu Wang for his contribution to this work from the theoretical perspective in a very helpful and competent manner. Parisa Memarmoshrefi, Ansgar Kellner, Youssef Shehadeh, Salke Hartung, Saleh Al-Shadly and Omar Alfandi for help on discussions and technical problems. Mehmet Gültas and Lennart Obermann for their discussion on the optimization problem, which pointed me to Keyu. Carmen Scherbaum and Udo Burghardt for their professional and kind support in the Telematics Group.

My Brother Eduard and wife Laura for proof reading my thesis from a computer scientist point of view. My parents for their unconditional and great support during my time as a student, this work is dedicated to them. And finally my family Katharina for her loving, patient and in words inexpressible support and my son Adam for enriching my life during my final PhD year.

Contents

Chapter 1
Introduction

Contents

1.1 Minimization of Wireless Interference

This thesis is concerned with the field of wireless multi-hop communication systems. The participating devices can be mobile or relatively fixed. They use wireless multi-hop technology to establish a network without deployment of wired backhaul links, however they can also be used to extend wired infrastructure networks in a fast, flexible and economical way. Computer networks in general have a limited capacity, meaning that a paths ability to transport data between one location and another is constrained by physical and logical properties of the network in terms of transmission of data units per time unit. Wireless multi-hop networks use two or more radio links (hops) to convey information between a source and destination. Because wireless links are subject to fundamental physical signal propagation effects, which make those links lossy and their signal omnipresent (broadcast nature), they require special medium access rules in order to realize successful transmissions. Due to these special properties wireless multi-hop networks are fundamentally limited with regard to capacity [41] compared to wired networks. In wireless networks in general the two circumstances loss of information and broadcast nature of radio signals can interact with each other. Radio spectrum is a shared resource. If two links in vicinity of each other transmit at the same time on the same frequency, their signal interfere, resulting in a collision. Transmitted information that suffers from a collision is considered lost and is not processed by the system. Therefore wireless networks are designed to prevent collision through a set of rules, the medium access protocol.

One medium access protocol concept is the carrier sensing and collision avoidance scheme used by IEEE 802.11 [79]. If a potential interference is detected by a transmitter, because the spectrum contains energy (channel is busy), it backs off from transmission and retries later. However medium access protocols cannot totally prevent collisions, they can occur. In any case interfering links in neighbourhood of each other result in either contention for the medium, in which case one link cannot be used for transmission at a given time or in the worst case a collision and a loss of all information. Both effects limit the capacity of a wireless network considerably.

In wireless multi-hop networks in particular two types of interference are prevalent. If two neighbouring devices (nodes) on different paths use the same channel, it is considered Inter Flow Interference (IrFI). If two nodes on the same path use the same channel, it is termed Intra Flow Interference (IaFI) [77]. Both effects are described in section 2.1.3 in more detail. Single channel single radio multi-hop networks suffer from above effects most. They can circumvent Inter-Flow Interference to a low degree and cannot prevent intra-flow interference at all, hence have a low nominal capacity. In this work we focus on IaFI as this type causes a capacity degradation on every consecutive hop with a uniform channel sequence and therefore is considered most severe.

If the capacity of a network path decreases due to interference and the amount of interference increases with network size, large networks can only offer poor rates of data transmission per unit of time with high rates of data loss and large amount of transmission delay. A network with such properties cannot offer acceptable quality of service to its users and applications. Therefore this problem is of high importance and has been in the focus of ongoing research for the past decade.

A general approach to remedy the interference problem is to relocate the transmission of data on neighbouring links to different non overlapping frequency bands of the spectrum, so called orthogonal channels. To realize that participating nodes must posses Multi-Interface Multi-Channel (MIMC) architecture. Nevertheless MIMC networks are subject to a network partitioning constraint, which means that neighbour nodes must have at least one common channel to ensure connectivity of the network. In literature a multitude of concepts for Channel Assignment (CA) is proposed under the connectivity constraint, most of which are centralized, graph theory based, computationally expensive and use strong assumptions, often not useful for practical applications.

One simple and practical concept is the assignment of a set of common channels to all nodes, the Common Channel Assignment (CCA), which we chose for our research in combination with the IEEE802.11s [88] draft standard and its routing protocol Hybrid Wireless Mesh Protocol (HWMP). To exploit the channel diversity of a network that has undergone some kind of channel assignment, a routing protocol must be able to form paths between source and destination in a way so that IaFI is minimized to ensure useful path and network capacity. Several routing metric concepts, as described in section 2.2.2, have been proposed in literature which provide awareness of IaFI in combination of HWMP with the goal to minimize it. However in this work we show that these concepts fall short of their goal, not because of lack of sophisticated routing metric design, but due to the intrinsic prevention of distance vector based HWMP to achieve that goal. The problem of a distance vector based routing protocol is the fact that optimization decisions during the path formation process are done by intermediary nodes which suffer from an effect we term Forward Path Unawareness (FPU). FPU prevents the protocol from considering all possible channel combinations along a path and results in IaFI-exposed links. As a consequence network capacity and quality of service suffer.

1.2 Thesis Contributions

The contribution of this work to the field of research is twofold. We propose a more radical alteration to ad-hoc distance vector based routing protocol HWMP than offered in literature. Our protocol collects a route record of all visited nodes along a path with all available links between the nodes, their channel number and channel metric, similar to the route discovery mechanism of Dynamic Source Routing (DSR) [40]. We term the protocol Route record based Distance Vector (RDV). It relocates the optimization problem from intermediary nodes of a path to the destination node, which computes the best sequence of channels with regard to the estimated link metrics and available channels at each hop. The first contribution however generates a complex non-trivial channel sequence selection problem, whose amount of sub-problems grows exponentially with the length of a path, if a naive algorithm is applied.

Our second contribution is the adaptation of a dynamic programming procedure called the Viterbi algorithm. This algorithm exploits our assumption that the Interference Range (R_I) with a grid node allocation is three hops and infers a second order Markov chain to solve the channel sequence problem in polynomial time. It does so by recursive solution of overlapping sub-problems and combination of their solutions to find a global optimum for a path. We design, specify and theoretically analyse the Viterbi algorithm, but reserve its implementation for future work. Nevertheless we approximate its function through the design of a greedy algorithm with low time and memory complexity and acceptable approximation error. Finally we gather evidence from extensive simulation which supports our hypothesis that RDV maintains a higher network capacity than HWMP combined with interference aware routing metrics. The field of research is complemented by this work with a different approach to distance vector routing and an effective solution to the channel sequence problem which results from it.

1.3 Thesis Organization

The thesis is structured as follows: In chapter 2 we provide a more detailed background for wireless multi-hop networks and discuss related work. We describe the basic concept of message relaying and which physical and logical constraints it underlies in a wireless environment. Following that we elaborate the fact that network capacity of wireless multi-hop networks degrades considerably with increase of network size and work out the causes for the capacity degradation. Further the chapter investigates which approaches have been undertaken in literature to improve network performance by introduction of MIMC architecture and which challenges it poses.

Chapter 3 specifies our system model such as node equipment, allocation, radio interface properties and channel assignment. It also gives examples of path formation with the base protocol HWMP and various routing metrics which aim at minimizing IaFI. Our examples show that IaFI minimization under our system model is inherently compromised by the path formation mechanism of HWMP, which is the core of our problem statement. Finally we state our hypothesis, that if the path formation mechanism of HWMP is altered to perform routing decisions on the destination node instead of intermediate nodes, with the help of the route record concept similar to DSR, it can better minimize IaFI and improve network capacity and performance.

In chapter 4 we specify the routing mechanism of RDV compared to HWMP. We find that RDV creates a channel sequence selection problem at the destination node. Further the chapter provides a theoretical analysis of the optimization problem and finds that it can be solved efficiently if the limited interference range assumption is exploited employing the Viterbi algorithm. Also an approximative Greedy channel selection algorithm is defined and analysed. We conclude this chapter with considerations for a practical implementation on real network nodes.

In chapter 5 follows a numerical evaluation of the Greedy channel selection algorithm in comparison to the Viterbi algorithm and a rudimentary approximative Diverse channel selection algorithm. We present an experimental evaluation of RDV in comparison to HWMP, which supports our hypothesis with regard to most evaluation metrics. Eventually we analyse RDVs packet delay performance, which is increased for certain traffic loads and derive items for future improvement.

In chapter 6 we conclude our elaboration and give an overview of future work.

Chapter 2

Background and Related Work

Contents

2.1 Background

To understand the position of this work and related work within the research area of wireless networks, this section provides a general background for the challenges of wireless multi-hop networks with respect to capacity degradation. Section 2.1.1 introduces to wireless multi-hop communication in general. Section 2.1.2 introduces to the origin of capacity degradation. Section 2.1.3 describes the causalities, mainly interference.

2.1.1 Wireless Multi-Hop Communication

In this section the very principal concepts of wireless multi-hop communication are described to give the reader a basic understanding of the field of interest in computer science.

In history the need to transmit information over distance without the exchange of a physical object that carries the information (which increases the rate of exchange), has been satisfied with the use of semaphores, before electric systems have been developed, for a long period of time. A semaphore can be a pair of hand-held flags [35], smoke signals or a heliotrope [17]. However semaphores rely on a Line-Of-Sight (LOS) and are limited by that fact. With the advent of electrical systems, the first electrochemical telegraphy experiment conducted by F.Salva in 1804 [92], was soon followed by electromagnetic telegraphs where only two conductors were necessary, but needed a binary code. Following the wire based telegraphy systems, radio telegraphy was introduced which allowed the exchange of information over large distances at minimal effort. Because communication range of radio signals is limited (the reason for this will be discussed below), messages were received and retransmitted by intermediary radio telegraph stations to bridge large distances. This brief historic perspective also applies to the work in this thesis, mainly the transmission of messages from one station to another using electro magnetic signals across large distances with multiple intermediate hops.

Figure 2.1 depicts the Open Systems Interconnection (OSI) model of Institute of Electrical and Electronics Engineers (IEEE) 802.11 based multi-hop networks. In this case we focus on IEEE 802.11s [88] draft standard, which forms a so called Wireless Mesh Network (WMN) on the link layer. The model represents internal functions of communication systems by partitioning into abstraction layers.

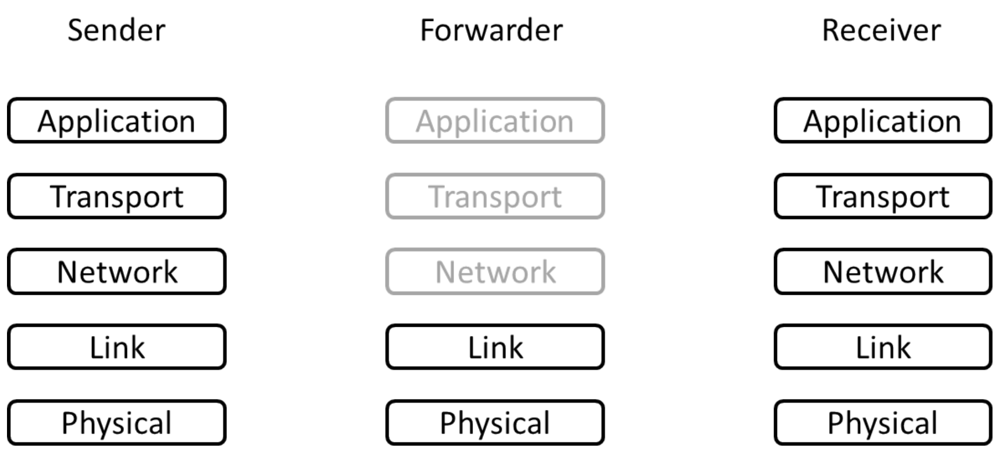


Figure 2.1: Open Systems Interconnection Model

When a sender needs to transmit a message to a receiver, its application layer hands it over to the transport layer which usually employs protocols such as Transmission Control Protocol (TCP) [18] or User Datagram Protocol (UDP) [68]. The transport protocol manages the delivery of various sized messages to the correct application. The transport layer hands it over to the network layer generally using Internet Protocol (IP) [14]. The network protocol manages the delivery of messages to the correct host system. It further hands it over to the link layer which encapsulates it in frames, addresses the frame with sender and receiver addresses, serializes it to a sequence of binary values and hands it over to the physical layer. The link layer manages the delivery of messages to the correct node. A node can have multiple host identities. The physical layer then uses digital modulation and spectrum access protocols to transmit the message over a certain radio frequency. An electro magnetic signal is a periodic waveform, which has certain properties such as amplitude, phase and frequency. The amplitude is the amount of electric voltage measured at the generating or receiving electronic circuit of the communicating nodes. The phase is the position within the wave cycle e.g. $180\,^{\circ}$ in relation to the origin $0\,^{\circ}$. Frequency is the amount of cycles of the waveform measured in cycles per second or Hertz (Hz). By variation of the three basic properties of a signal the sender encodes the digital information which needs to be transmitted. This encoding of the two symbols 0 and 1 is what is understood under modulation. Spectrum or medium access protocols are rules for the access of the channel at the correct time, or frequency so that the transmission is successful and fair for all participants. Intermediate nodes receive and demodulate the message on physical layer, decide on the link layer to which next intermediate (or destination) node it should be forwarded to and retransmit the message. Notice the higher layers above link layer are not active on forwarding nodes in case of 802.11s, but can also be formed on network layer in other approaches. When the receiver captures the message on its physical layer, it hands it over to the upper layers for de-serialization, de-capsulation and final delivery to the correct application.

Wireless communication systems can use different physical media such as sound or electromagnetic waves. Here the focus is on electromagnetic waves in the practical range of a few hundred Mega Hertz (Mhz) to a few Giga Hertz (Ghz) [48]. The physics of radio propagation underlies several effects which need to be considered. The first effect is the decrease of signal strength with increase of distance between sender and receiver. When a signal is radiated its mean power is distributed over a volume around the sender. Equation 2.1 describes the loss of signal power in free space with a theoretical isotropic antenna according to [31].

$$P_r = \frac{P_t}{4 \cdot \pi \cdot d^2} \tag{2.1}$$

The received power P_r is proportional to the distance according to the inverse square law. The transmitted power P_t is emitted into all directions and the absorb-able energy per unit area at a given distance is equal to the area of the surface of a sphere, hence the constants $4 \cdot \pi$. Notice the equation above is a strong simplification of the real effect and does not take into account the properties of antennas and used wavelength.

The second effect is interference from other sources such as galactic radiation and other nodes and transmitters of other technologies using the same frequency in vicinity of the receiver. This disturbance can interrupt, obstruct or otherwise degrade or limit the effective performance of the electrical circuit of the physical layer of the receiver [83]. Interference is experienced as noise at a receiving node. Interference in wireless networks can be modelled in two ways, the protocol model or the physical model. The protocol model assumes that a data packet is transmitted successfully if the receiver is within Communication Range (R_C) and if no other nodes within R_I of the receiver transmit at the same time. The physical model states that a data packet is transmitted successfully if, at the receiver, the signal power level in relation to interfering noise from other transmissions is above a certain threshold. This relation is termed Signal-to-Noise Ratio (SNR). The maximum rate at which information can be transmitted in the presence of noise is limited as initially described by Claude Shannon [76]. Another term used to describe the relation of signal power and discriminate between noise from the background and interference by other links of the same network is Signal-to-Interference plus Noise Ratio (SINR) as defined in equation 2.2, where the received signal power P_r is weighted by I, the interference power by other simultaneous transmissions and N background noise.

$$SINR = \frac{P_r}{I + N} \tag{2.2}$$

A higher SINR represents a higher channel capacity and allows to successfully transmit more information per unit of time compared to a low SINR. It is necessary to point out that the modulation in wireless communication is related to SINR. Modulation can vary in density, which means one can model few bits per second on a radio signal or multiple by combining basic modulation techniques such as Amplitude Shift Keying (ASK), Phase Shift Keying (PSK) and Frequency Shift Keying (FSK) or multi carrier modulation [74]. With a high SINR a high bit rate modulation scheme can be used, as SINR decreases either due to increase in distance or in interference or noise the modulation scheme must be adopted to a lower bit density in order to successfully transmit data. The reason for this dynamic modulation necessity lies in the fact that high bit rate schemes contain many states that encode a certain bit sequence. The more states are encoded the closer they are to each other. Noise and interference or greater distance can make the states undistinguishable and create packet corruption. Therefore low bit rate modulation schemes are more robust and suitable for lower SINR than high bit rate schemes.

The third effect is multi-path propagation of the signal between sender and receiver. Multi-path propagation is the result of effects such as reflection, diffraction and scattering of radio signals among others [3]. An emitted signal can arrive at the receiver through multiple paths, such as directly in LOS and indirectly without

LOS because its direction and velocity can be obstructed by above effects. The result is a degradation and time dispersion of the signal which results in Inter-Symbol-Interference (ISI) and increased bit error rate.

One can describe the above effects in relation to each other as follows. A radio transmitter emits radio magnetic waves which represent (modulate) certain information at a given energy level P_t. At the receiver a degraded form of the signal is absorbed P_r. The amount of degradation depends in free space on the distance, wave length, antenna properties and in a real world environment also on interference and multi-path propagation. Transmission errors on wireless links are considerably more frequent compared to wired links, hence the physical and link layer must compensate this with potent error detection such as Cyclic Redundancy Check (CRC) [67] and error correction such as convolutional codes and potential frame retransmission.

This thesis aims at increasing network performance through increasing SINR by reducing noise from interference. To understand how interference is reduced by our concept, it is necessary to comprehend the types of interference in a wireless multi-hop network and how it is created. Interference can be generated from three different sources. The first source is external to the network and can be the result of neighbour networks and other technologies and appliances using the same frequency which create background noise. The second and third sources are internal to the network and result from inter-flow interference and intra flow-interference. Section 2.1.2 describes both types with an example.

In chapter 3 a comprehensive example is provided which describes the mechanism for HWMP and how it handles interference.

2.1.2 Capacity Degradation of Wireless Multi Hop Networks

The problem of capacity degradation for wireless Carrier Sense Multiple Access (CSMA) based multi hop networks (such as 802.11) has been studied from a theoretical perspective by [33] and [51]. They notice that the observed capacity of deployed networks is significantly below the theoretical optimum. A geometric analysis of the available bandwidth of one common channel to a node of a network is done by [33]. They assume a non interference protocol, possibly achieved through spatial separation and power control as proposed by [7]. The authors use both interference models, protocol and physical for an examination network capacity. They find that the upper bound for available throughput for a node that chooses its communication partner randomly under the protocol model is as given in equation 2.3.

$$T(n) = O(\frac{W}{\sqrt{n \cdot log \cdot n}}) \tag{2.3}$$

Here n is the total number of nodes and W is the available channel bandwidth in Mega Bit per Second (Mbps). The "big o" notation is used to describe that the throughput $T(n)$ available for any node for a random communication partner has an upper bound if n grows towards infinity.

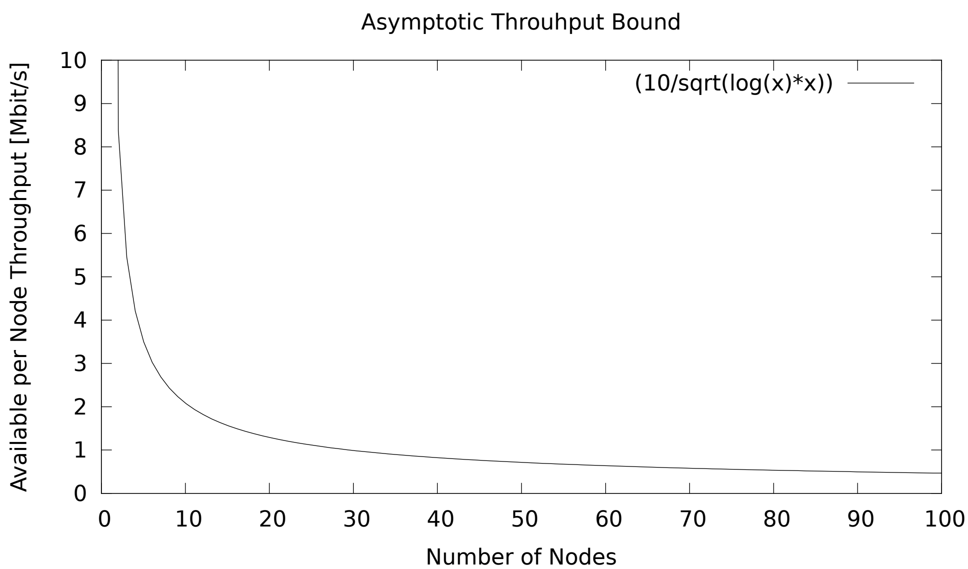


Figure 2.2: Asymptotic Throughput Bound

Figure 2.2 shows the asymptotic bound for the available bandwidth, where the ordinate (y-axis) represents throughput in Mbps and the abscissa (x-axis) represents the network size. For example, if all nodes are equally set up and their common channel offers a bandwidth of 10 Mbps, than for a network size of 100 nodes, the available per node throughput for random communication partners is up to 0.7 Mbps. Notice very small networks this upper bound does not fit, it is however correct for larger networks as described by the right hand part of the diagram.

In practice interference on a common channel is challenging to prevent since R_C is smaller than R_I, hence we note that the non interference assumption of the authors is strong, serves rather a theoretical purpose and cannot be used in practice.

In the works of [53], [25] and [52] the authors analyse ad-hoc routing protocols and find that the observed capacity is up to 50 times smaller than the apparent channel bandwidth. For the above example the per node throughput would be about 0.2 Mbps, less than three times smaller than equation 2.2 predicts. The authors of [53] use a setup of 50 nodes with single and multiple interface and one common channel to evaluate the performance of routing protocols such as DSR [40], AODV, Destination-Sequenced Distance-Vector (DSDV) [65] etc. The authors of [25] use a similar setup to compare

AODV and DSR. As a consequence the set traffic load for routing protocol studies is a small fraction of the channel bandwidth in order to prevent transmission failures as a result of congestion losses. As one can see the asymptotic per node throughput in theory and practice diminishes towards 0. Following this insight the general implication by the authors is to either limit the size of wireless multi hop networks or to keep communication local in larger networks.
For the case of wireless mesh networks, which are a special case of ad-hoc networks, where the communication pattern is mostly not between random peer nodes, but rather between user nodes and gateway nodes, the theoretical upper bound for available per node throughput $T(n)$ is as described by equation 2.4 [41].

$$T(n) = O(\frac{W}{n}) \tag{2.4}$$

They use to concept of collision domain to examine the transmission capacity restrictions of WMN A collision domain is defined for a link that needs to transmit data. It is a set of neighbouring links that have to be inactive for the given link to transmit successfully. The authors define the term bottle neck collision domain, which is the collision domain that has to transmit the most traffic in the network. In WMNs this is usually formed as a contention hots spot around a gateway node. This shows that the available per node throughput resembles the single hop case and is considerably worse than the general results of [33] for random ad-hoc communication patterns. The reason for the decreased performance of WMN compared to random ad-hoc Networks is the fact that the hotspot around a gateway node throttles the throughput for every node of the network. In fact the authors of[41] argue that the capacity of wireless mesh networks in practice is 2-3 times lower than the single hop case which is not expressed by the big o notation. In above example the nominal per node throughput would be about 0.05 Mbps. This in general is close to the observations of [75], which led the authors of this work to investigate this topic initially.

2.1.3 Causes for Capacity Degradation

There are several reasons for the reduced performance of wireless IEEE 802.11 based multi hop networks if one common channel is shared by an increasing number of nodes.

Interference is identified by [37] as a major factor for capacity degradation. They propose a method for the computation of capacity bounds by modelling interference with a conflict graph. As a result they derive the implication that routing protocols with interference aware metrics can benefit the performance of wireless multi hop networks compared to shortest hop metrics. Such a routing protocol can find longer or more circular paths from a source to destination by avoiding interference prone links.
In the work of [90] two kinds of interference are identified. The first kind of interference

results from the fact that a packet flow through a multi hop network not only consumes the channel bandwidth at the nodes along its path from source to destination, but it also contends for channel bandwidth with neighbouring nodes of the path. Their example is reproduced in the following two figures.

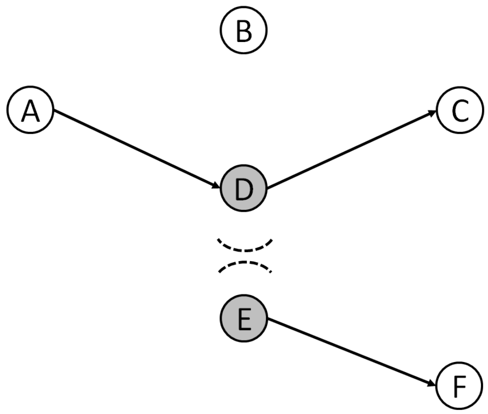


Figure 2.3: Inter Flow Interference

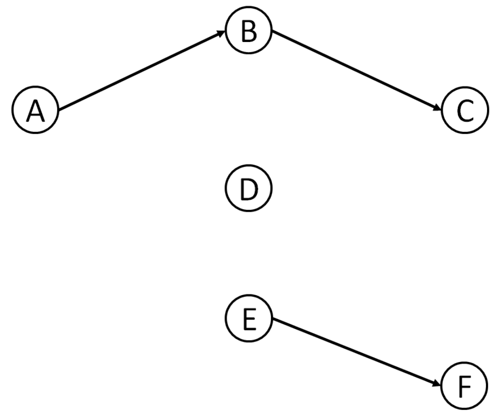


Figure 2.4: No Inter Flow Interference

Figure 2.3 illustrates that the flow along path $A-D-C$ contends for the channel bandwidth with the flow along path $E-F$ because both paths are in direct neighbourhood and within interference range. In figure 2.4 on the other hand the flow along path $A-B-C$ does not contend for the medium with the flow along path $E-F$ under the assumption that node A/B are out of interference range of node E. This type of interference where nodes of two different flows along neighbouring paths contend for the same channel is termed Inter Flow Interference and will be referenced as IrFI in the following sections. In other terms, let path P be a candidate [84] route for a desired data exchange between two nodes. IrFI is caused by interference between a wireless link on path P and other links that are not on P and currently used for transmissions. The second type of interference is observed if nodes on the path of the same flow compete for the same channel bandwidth, which is always the case in a Single Interface Single Channel (SISC) network. However in a network with MIMC architecture it can be avoided because non overlapping channels allow to successfully transmit data at the same time. The following figure describes interference in both cases.

Figure 2.5 shows a flow along path $A-B-D$ which reuses channel 1 for the relaying of packets. Figure 2.6 shows a flow along path $A-C-D$ which uses a different channel for packet relay. This type of interference within the same flow is termed Intra-Flow Interference and will be referenced as IaFI. In other terms IaFI is interference caused by a wireless link on path P and other links on P. IaFI increases the bandwidth consumption of a flow at every hop where a channel is reused, it increases the packet delay and reduces flow throughput dramatically with each additional channel uniform

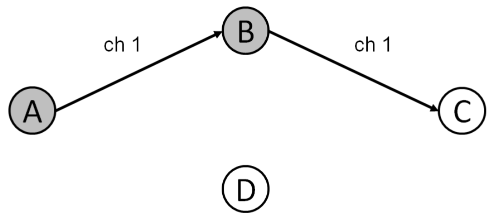


Figure 2.5: Intra Flow Interference

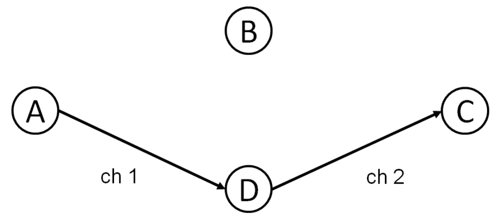


Figure 2.6: No Intra Flow Interference

hop. The authors propose that routing metrics should consider both types.
As a summary a simple example shall give an intuitive understanding for the general IEEE 802.11 capacity degradation problem. The peak link layer data rate of e.g. IEEE 802.11g is 54 Mbps, the used modulation type is 64 QAM. Multiple access rules and error handling mechanisms reduce the data rate to about half i. e. 27 Mbps. As an intermediary result two nodes which are close to each other and have no neighbours, can communicate with about 50% of the peak data rate. Increasing distance between sender and receiver reduces the received power level at the receiver by the inverse square law, which reduces SINR, this leads to the necessity to adapt more robust and less bit-dense modulation schemes and reduce the link layer data rate further to for example 12 Mbps using QPSK. This means that the available throughput shrinks to 6 Mbps, 11% of the peak data rate. If a neighbouring transmission is added to the scenario, the channel needs to be shared and in the best case the available bandwidth is shared fairly, the available bandwidth shrinks to 3 Mbps, 5.5% of the peak link layer rate. If the transmission is longer than the above mentioned two nodes and packets need to be relayed and the channel for forwarding and receiving of packets is the same, the bandwidth is divided by 2 again to 1.5 Mbps, 2,75% of the peak link layer rate. Every additional neighbouring transmission and channel uniform hop within the same flow strongly reduces the bandwidth further to a small fraction of the 54 Mbps. The issue is that wireless multi hop networks per definition consist of multiple simultaneous flows and multiple packet relays if under utilization. Therefore single channel multi hop networks due to above reasons cannot offer more than small fractions of the peak link layer data rate.

2.2 Related Work

The capacity degradation problem of wireless multi-hop networks can be summarized with the facts that single channel networks scale poorly because of interference. Originally IEEE 802.11 is designed primarily so that nodes use one half duplex interface in a single hop scenario. As a consequence in order to maintain connectivity in multi-hop scenarios one common channel must be used.

After the capacity degradation was studied thoroughly by the research community focus turned to resolving the issues by utilizing multiple channels for IEEE 802.11 based multi hop networks. The concepts can be ordered in two subcategories. The first category concentrates on the assignment of channels to links so that interference is minimized. The second category focuses on interference aware metrics, if a channel diverse topology already exists. Channel assignment concepts again are divided into to subcategories the first proposes the use of a single radio interface and multiple non overlapping channels, we term this category as Single Interface Multi Channel (SIMC) concepts. The second subcategory proposes MIMC concepts.

This thesis is focused on path formation (routing) mechanisms and interference aware metric calculation. However initially it is necessary to determine what kind of CA concept shall be assumed for our system model. In the following sections we give an overview of different CA scheemes, than we choose one for our work. Eventually we discuss some interference aware routing metric concepts before we state our problem in chapter 3.

2.2.1 Channel Assignment in Wireless Multi-Hop Networks

The authors of [6] propose a channel hopping link layer protocol for IEEE 802.11 with the goal to increase single-interface multi-hop network capacity, by exploiting channel diversity of orthogonal (non overlapping) channels. In their concept nodes that need to communicate with each other switch to overlapping channels, while disjoint communications tend not to overlap. Each node creates a hopping schedule for each channel. Each node then broadcasts its schedule to neighbour nodes and updates it periodically to adapt to varying traffic patterns. The concept implements a distributed rendezvous and synchronization mechanism in order to reduce interference and increase per node throughput. Their results show that it is possible to increase the overall network capacity for multi hop networks with a single interface and multiple channels. They use DSR for route discovery and a network size of 100 nodes, where 10 node pairs (selected randomly) engage in (UDP) communication. They show that short paths (1-5 hops) their SIMC approach does not have any advantage over standard IEEE 802.11 SISC. For longer paths (5-18 hops) however their approach maintains a low available throughput, while SISC diminishes to 0. Overall network capacity under high load (many simultaneous transmissions) is increased by SIMC over SISC. The work of [58], [38], [59] and [86] falls into this category, they share the goal to reduce collisions and thus interference by using multiple channels and a single radio per node.
The drawbacks of the SIMC approach in general are the necessity for tight time synchronization across nodes, complex topology control mechanisms in order to maintain network connectivity and switching delays of IEEE 802.11 hardware. Further SIMC

concepts cannot address the IaFI problem, since simultaneous reception and transmission is not possible with a single half duplex radio interface. The fact that the channel hopping concept of [6] suffers under a significant throughput decline for short path lengths underlines the problem. If a node has multiple interfaces, combined with multiple orthogonal channels it can receive on one channel and forward on another. MIMC concept therefore can exploit channel diversity not only for parallel (neighbouring) paths, but also for consecutive hops of the same path. This can potentially increase network capacity of multi hop networks significantly compared to SIMC concepts.
MIMC concepts face a multi problem challenge. The first problem is the decision which channel should be assigned to a link between two nodes in order to improve network capacity by creating a channel diverse topology under the constraint of maintaining network connectivity. We refer to this problem as the CA-Problem.
The second problem concerns the decision which path of the potentially many available in a multi-channel multi-hop network between a source and destination node to chose in order to achieve a high individual flow performance and high overall network performance by exploiting a channel diverse topology, which is created by a solution to the CA-Problem. We refer to this as the Routing Problem. It is necessary to stress that both problems are interrelated as CA can depend on routing and vice versa.
There is a multitude of research concerned with MIMC concepts, we divide into centralized and distributed concepts, which in turn can have either static or dynamic CA. The static type separates the routing and CA-Problem. It assumes that CA can be performed independently of routing decisions, by creating a channel diverse topology, which is then efficiently exploited by adapted routing protocols with channel diversity aware routing metrics. The dynamic type changes channels over time depending on routing decisions or indirectly on measured interference levels or other parameters. Table 2.1 shows a classification of work in literature on channel assignment and routing.

Type	Subtype	Static	Dynamic
centralized	Common CA	MR-LQSR [29] MUP [1] BFS-CA [71]* MESTIC [78]* Net-X [11], [49]	COMTAC [60]*
	Varying CA	C-HYA [72] CLICA [54] MESTIC [78]* INSTC [84] B-CA [93] CTA [82] ILP [20] RCL [2] BS-CA [46] MCI-CA [13] MC-CA [5]	LA-CA [72] PD-CA [46] BFS-CA [71]*
distributed	Gateway Oriented CA	DMesh [24] IA-CA [42]	D-HYA [73] COMTAC [60]*
	Peer Oriented CA	PCU-CA [50]*	PCU-CA [50]* JOCAC [69] SC-CA [89] SS-CA [45] D-CA [20]

Table 2.1: Channel Assignment and Routing in MIMC Networks

*Notice: * marks MIMC schemes which share properties of the various subtype, which means they are hybrid approaches, which for example assume a common channel for topology control and varying channels for channel diversity.*

The centralized type employs a central algorithm to optimize channel allocation and routing decisions by taking into account various input parameters such as node position, number of orthogonal channels and radio interfaces, traffic profiles, interference measurements or estimations and other constraints such as network connectivity. The general objective of centralized algorithms is to increase network throughput, decrease delay or reduce packet loss all of which are interrelated. Most centralized algorithms are graph based and use approximation algorithms so solve the Non-deterministic Polynomial-time hard (NP-hard) [71], [72], [54] problem of assigning channels in arbitrary graphs. Centralized types assume that knowledge of the whole network (its parameters) is available at one place and that output (channel assignment/routes) is distributed to every node after computation. Centralized concepts are further distinguished between Common Channel Assignment (CCA) and Varying Channel Assignment (VCA).

CCA is a trivial CA scheme where all radio interfaces are assigned the same set of channels [78]. CCA has no network connectivity or network partition issue, it increases network performance by exploiting channel diversity. It is limited by the fact that channel diversity is limited by the number of radio interfaces of the nodes. If more orthogonal channels than radio interfaces are available VCA can potentially create higher channel diversity and therefore further increase network performance.
VCA concepts can assign different channels to radio interfaces. The main challenge of VCA concepts is to create an optimal channel diverse topology while maintaining network connectivity which comes at the cost of increased concept complexity and reduced practicability as we will show in the literature review below.
With the distributed type, each node runs its own copy of the algorithm, which assigns channels to its interfaces [77]. Input parameters are available locally through measurement and exchanged with neighbour nodes. The parameters used are for example traffic load, SINR or number of links on a given channel in R_I. This type needs a coordination mechanism between nodes for parameter exchange and channel assignment negotiation in order to ensure connectivity.
Gateway Oriented (GO) distributed CA schemes use a network gateway to simplify channel assignment. The CA is directed from regular nodes towards gateway nodes under the assumption that the traffic profile is biased towards latter. Their main drawback is the inability to provide optimized CA for peer to peer communication within a multi hop network.
Peer Oriented (PE) schemes are more general regarding the assumed traffic pattern and provide CA for all types of traffic. Their main challenge is the coordination of channel assignment since no topological ”anchor” such as a gateway node can be assumed.
Below we provide an exemplary description of recent work on MIMC CA and routing schemes. We will present centralized VCA, distributed GO and PE schemes and finally cover centralized CCA concepts.

2.2.1.1 Centralized Varying Channel Assignment

The authors of [93] propose a centralized static VCA scheme termed Better Channel Assignment (B-CA). It is graph based and uses a heuristic algorithm. It initially creates a 2-connected graph using a redundant tree algorithm by [56] which is robust against single node failure, thus prevents a certain degree of network partitioning. Then it assigns orthogonal channels evenly across the graph by taking potential interference from neighbour nodes and their assigned channels into account. As input it takes node number, node position and set of channels. It assumes two radio interfaces per node and as constraint 2-connectedness. As output it provides a 2-connected topology with low interference-CA.
Its advantage is the simple solution for connectivity by applying the redundant tree

algorithm for topology creation, where an internet gateway node is the tree root. It offers a theoretically simple implementation in one place.
The disadvantages are the dependency on node position which needs to be collected and centrally put in to the algorithm as well as the fact that the solution needs to be distributed after computation. Further its static nature does not allow adaptation to topology changes or reaction to traffic profile alterations. Also this approach does not consider routing and does not provide simulation or test bed evaluation.
The authors of [72] propose a centralized dynamic VCA scheme termed Load Aware Channel Assignment (LA-CA). With LA-CA routing and CA co-depend on each other. They assume that node positions (also referred to as unit disk graph) and traffic load estimations between any two nodes are known a priory. The initial link load estimations are used to assign channels to links between nodes initially. After that the routing algorithm (not specified) updates the end-to-end routes based on available channel capacity. The reaction of routing to channel assignment changes end-to-end traffic rates which in turn change link load estimations which cause channel reassignment as long as link loads are greater than link capacity. They prove that the CA-Problem is NP-hard by reducing the Multiple Subset Sum Problem to the CA-Problem [15] and use a greedy heuristic algorithm to solve the problem, which output objective is to make sure that a links capacity can serve its aggregate traffic load from every node using it. They demonstrate an improved network performance compared to SISC by simulation in Network Simulator 2 (NS-2) [62] and test bed implementation.
This approach demonstrates feasibility by using a simple heuristic algorithm and displays improvement in experiments. Its general drawbacks are the dependency on traffic estimations, node positions and the fact that it is a centralized solution to a distributed problem, which is theoretically sound but impractical.
There exist hybrid schemes which use statically assigned channels on certain radio interfaces and dynamically assigned channels on others such as Breadth First Search Channel Assignment (BFS-CA) [71].
Centralized VCA concepts represent the majority of research for MIMC networks with a multitude of publications between 2004 and 2013. They have in common, being executed at one place, using similar input parameters such as a unit disk graph, channel parameters, traffic profiles and connectivity constrains. They vary in the interpretation of interference model, assumptions and applied approximative algorithms to derive their solutions. Some schemes consider routing others don't. Please see table 2.1 for a further overview of concepts.

2.2.1.2 Distributed Gateway Oriented Channel Assignment

A distributed dynamic CA-scheme is proposed by [60]. It is termed Cluster-based Multipath Topology control and Channel assignment (CoMTaC). It assumes the existence

of Internet Protocol gateway nodes. To solve the connectivity problem it proposes a two phase approach. In phase one, during network initialization the network is split into clusters where gateway nodes represent cluster heads. A default radio interface with a default channel is used for intra-cluster connectivity. A second (non-default) radio interface is used to increase connectivity within the cluster. For border nodes the second interface is used to establish inter-cluster connectivity. In phase two the CA is modified to reduce interference. The default channel is measured for utilization and signal quality by cluster members and reassigned by the cluster head if a better, less busy channel is available. Because of this property we see this concept as a hybrid between a distributed dynamic and centralized (cluster head being central) dynamic common channel assignment scheme. The non-default channels are also measured by cluster members for link layer queue length and reassigned by the cluster head in a prioritized manner with the objective to reduce interference within the cluster, to improve inter-cluster connectivity and to reduce interference in vicinity to the gate way nodes. CoMTaCs advantage is the simple solution for topology control (connectivity problem) by using a default common cluster channel and the efficient interference estimation by using the link layer queue length. However its dependence on a gateway node limits optimization to only gateway oriented traffic patterns.
There are several other concepts which provide similar solutions such as the non-cluster based Distributed Hyacinth (D-HYA) [73] and the static DMesh [24].

2.2.1.3 Distributed Peer Oriented Channel Assignment

A more general approach to assign channels in a MIMC multi hop network is to relax the assumption that gate way nodes are present and that traffic is biased towards those nodes. [50] propose a Probabilistic Channel Usage based Channel Assignment (PCU-CA) scheme which is peer node oriented. In their concept nodes have two kinds of radio interfaces. The first kind uses a fixed channel for a (configurable) period of time. The fixed channel is determined by local measurements of channel utilization. A node measures how busy its fixed channel is and it evaluates the measurements of its one hop neighbours, which are distributed in "hello" messages. If the channel utilization is high, the node decides wo switch its periodically fixed radio interface channel to a less busy channel. To prevent oscillation between channels for the periodically fixed interface, the node will only switch with a certain probability.
The second kind of interface is dynamically switch-able at any time. A node maintains a queue for every channel and switches with its dynamic interface to channels of the fixed channels of its neighbours to forward the packets in its queues. Because the fixed channels of neighbours dictate to which dynamic channels the second interface will switch, the periodically fixed radio interfaces determine the channel assignment. An adaptation of DSR is used for routing. An interference aware routing metric is

used which aims to exploit the available channel diversity created by PCU-CA. The authors present a simulation based evaluation in Qualnet [85] with improvements in network performance compared to SISC. We see this concept as a hybrid of dynamic and static distributed peer oriented network because of the two kinds of channel assignment strategies.
The advantage of the approach is the CA algorithm which is based on local measurements of channel utilization without loss of generality regarding dependence on gateway nodes. PCU-CA further solves the channel oscillation problem by making the channel switching probabilistic. The limitations are the cost of switching delays for the dynamic interfaces and the fact that a probabilistic switching method for the fixed interface does not always guarantee reduction of interference and improved performance.
Other concepts exist in this category. [69] propose Joint Optimal Channel Assignment and Congestion Control (JOCAC) a CA-scheme from perspective of TCP congestion control that applies a distributed utility maximization algorithm. [89] propose Superimposed Code based Channel Assignment (SC-CA), a scheme which applies code theory to select channels with low interference. [45] propose a Self-Stabilizing Channel Assignment (SS-CA) algorithm which greedily selects a channel with low interference, coordinated by a stabilization protocol that applies mutual-exclusion operations for coordination of channel switching.

2.2.1.4 Centralized Common Channel Assignment Concepts

The authors of [1] propose a link layer protocol termed Multi-Radio Unification Protocol (MUP) which manipulates channel selection on an intermediate node in order to reduce interference, optimize spectrum utilization and increase overall network performance. The main idea is to use a common set of channels on all nodes. This eliminates the connectivity problem of VCA schemes which rely on non-trivial CA-coordination. Further it excludes the switching delay and coordination problem of dynamic CA-schemes. MUP does not focus on optimal CA, but attempts to efficiently exploit channel diversity of a simple CCA pre-assigned network. In figure 2.7 a simplistic topology with two orthogonal channels A, B consisting of node n_1 (traffic source), n_2 (intermediate) and n_3 (traffic destination) is depicted. All nodes are MUP capable.

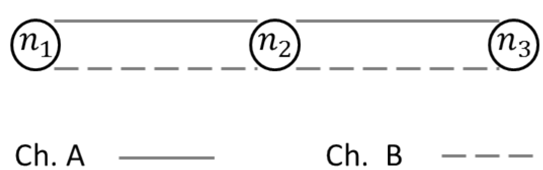


Figure 2.7: Multi-Radio Unification Protocol

When node n_1 sends data to node n_3 via n_2 it measures the Smoothed Round Trip

Time (SRTT) on both channels and chooses the channel with minimal SRTT, which indicates low channel activity, thus low interference. We assume in this example node n_1 chooses channel A. When node n_2 receives the packets from node n_1 destined for node n_3 it also measures SRTT on both channels and chooses the less busy channel. Since channel A is occupied by n_1s transmission node n_2 chooses channel B as the forward channel.

The authors propose the location for MUP between the network layer and the mac layer of a nodes protocol stack in order to simplify the design and prevent any changes to established protocols. They conduct an evaluation employing NS-2 with a 12 node grid topology. They show significant throughput increase and delay decrease compared to SISC. The routing algorithm used is Ad-Hoc On-Demand Distance Vector Routing (AODV).

The proposed concept is trivial in that it aims at increasing multi hop network capacity without the attempt of finding a globally optimal CA solution. As was mentioned in section 2.2.1.1 a globally optimal CA belongs to the group of NP-hard problems and even with the employment of heuristic algorithms computationally expensive [37]. It is further compatible with legacy non MUP IEEE 802.11 architecture because it does not change existing medium access or routing protocols. The authors exploit a simple CCA by locally deciding which channel should be used for the next hop of a packet.

The drawbacks of this scheme are the simplistic link quality metric SRTT which does not consider channel bandwidth, error rate or previous hop channel diversity. The concept also ignores the routing protocol, which is AODV and with its shortest hop routing metric does not consider congested hot spots. Channel diversity and routing thus are decoupled which must lead to suboptimal routing decisions.

The authors of [29] propose a routing protocol based exploitation of CCA MIMC networks. Compared to MUP their approach is more sophisticated in that it not simply chooses a less busy link based on round trip time measurements between two nodes, but it introduces a routing metric which is aware of channel diversity. DSR [40] is used as a base for proposed Multi Radio Link Quality Source Routing (MR-LQSR) for route discovery and maintenance. It is extended with awareness on a links error rate, available bandwidth and a weight system to account for available channel diversity.

Contrary to DSR which uses a simple shortest hop path metric and to Expected Transmission Count (ETX) [22] which estimates the transmission time by taking retransmissions into account, they propose Weighted Cumulative Expected Transmission Time (WCETT) discussed in more detail in section 2.2.2. MR-LQSR collects the metric values within a route record between source node S and destination node D by broadcasting a Path Request Packet (PREQ) frame. When a PREQ frame arrives at D, it answers with a Path Reply Packet (PREP) which contains the route records. When S receives the PREP frame it computes the metric for the route to D according to above equations. If S sends data packets to D it provides the route record within the

packet, which is used by intermediary nodes to forward the packet to the destination. With a DSR based routing algorithm intermediary nodes only know their neighbours and do not store routing table entries for the destination. All necessary information for forwarding is provided by the source, hence the term source routing algorithm. The authors implement the protocol between the network layer and the link layer and extend DSR PREQ and PREP frames with an Expected Transmission Time (ETT) and channel number field. They conduct a 23 node test bed evaluation where all nodes have two radio interfaces tuned to orthogonal channels. They demonstrate a network performance increase compared to a SISC base line scenario and a MIMC scenario with the ETX metric. The improvement is significant for short paths and limited for longer paths or heavy network loads.
The advantage of MR-LQSR is the fact that a complete route record is collected, which is used to compute a metric which is able to consider IaFI because all nodes and the used channels between them are known. Further MR-LQSR considers both the link error rate and available bandwidth to estimate how busy a given channel is. The disadvantage of MR-LQSR is the fact that its metric only considers the channel with the highest IaFI as the bottleneck channel. It therefore cannot give an estimation of how long a packet between source and destination will travel. The attempt to use the left part of equation 2.9 to estimate an ETT for the whole route is with a weight β is unrealistic.

2.2.1.5 MIMC Discussion

In previous sections it has been shown that IEEE 802.11 based multi hop networks with SISC architecture suffer from low performance and due to low scalability. The causes have been identified as IaFI and IrFI. The state of the art concepts have been ordered in categories and chosen ones have been presented and discussed. It can be said that there exists a trade off between potential channel diversity, hence less interference and better performance and complexity therefore less practicability and higher cost of optimization of the proposed schemes as figure 2.8 depicts.

Centralized VCA schemes for instance offer potentially highly channel diverse, well scaling multi hop networks at the cost of complex, computationally expensive solutions with impractical assumptions such as known network graphs and their parameters.
Centralized CCA schemes on the other hand offer less channel diversity due to their fixed scheme, but are more simple and therefore practical. They do not have any topology control issue, no switching delay problem and do not need complex CA algorithms. Their main challenge is the optimization of the path selection problem which reduces the focus on the routing protocol and the routing metric.
Distributed gateway oriented CA schemes offer more potential channel diversity than CCA schemes, but require more complex channel parameter measurements and mech-

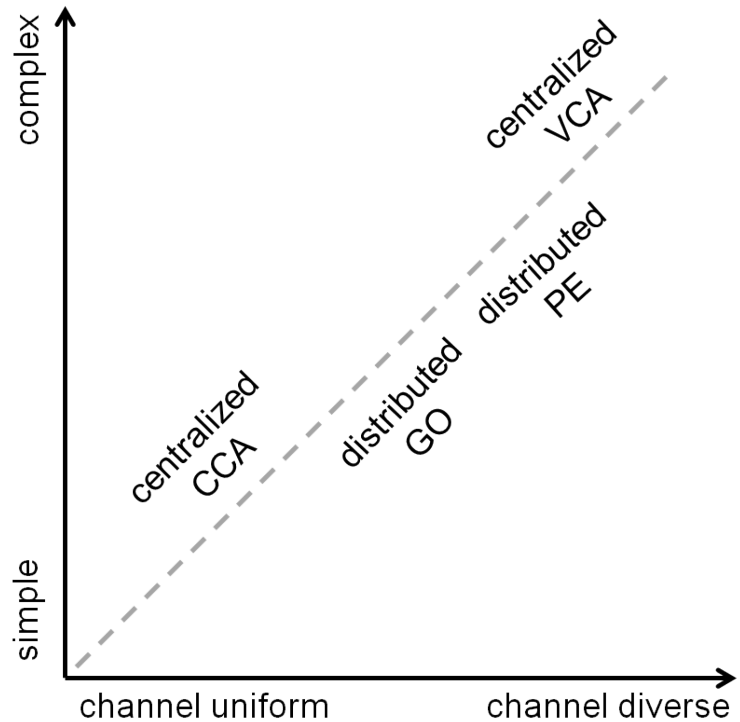


Figure 2.8: Complexity vs. Practicability of CA Schemes

anisms for channel switching. Distributed peer oriented CA schemes offer even more potential channel diversity because they do not assume gateway node existence, but require more complex topology control and channel reassignment mechanisms compared to gateway oriented schemes.

As can be seen in table 2.1 the area of CCA schemes has had relatively low attention by the research community due to the fact that it offers relatively modest channel diversity. However the reduced complexity compared to VCA and distributed concepts make CCA the most practical and relevant. In work we chose CCA for our system model which is defined in section 3.1. This allows us to focus on the routing problem primarily in combination with the distance vector based routing protocol HWMP. In section 3.2 it is shown that a distance vector based routing protocol has a weakness in exploiting channel diversity of networks that use CCA due to how its path formation mechanism is designed. In this work we set the goal to remedy this weakness by changing the path formation mechanism to allow better exploitation of channel diversity.

2.2.2 Routing in Wireless Multi-Hop Networks

Routing in wireless multi-hop networks is a well studied research area. To understand how packet flows between sender and receiver are established we briefly explain path discovery mechanisms of wireless multi-hop networks. When a flow between two nodes needs to be established, the network layer, or in case of HWMP the link layer initiates a path discovery mechanism to find an optimal sequence of intermediate nodes. This path discovery is realized by a routing protocol. A routing protocol is a set of rules for intermediate nodes (routers) which dictate how to disseminate information about available links which enables them to discover and select a route between two terminal nodes. Routing protocols can be categorized into two types.

The first type is link state routing, based on initial work of McQuillan [55]. Its basic concept is that every node in the network constructs a graph (map) of the whole network through cooperative dissemination of information on available links to neighbour nodes. Using this graph every node calculates the best path to every possible destination from its point of view. The calculation is based on cost of links which express link characteristics such as signal quality etc. Variants of the Dijkstra [27] algorithm are predominantly used for least cost path calculation. An example for link state routing is Optimized Link State Routing Protocol (OLSR) [21]. It is important to note that any topology change, for example a broken link at a node, must be advertised to all other nodes in the network.

The second type is distance vector routing, which is based on the initial work of Bellman [8], Ford [30] and Moore [57]. The general idea is that nodes do not know the complete path to a destination, hence no complete network graph exists. Instead nodes know the direction (link for the next hop) and distance (cost) to a destination. They maintain a list (vector) of least cost directions for every node. An example of distance vector routing is Routing Information Protocol (RIP) [34]. Notice that any topology change at a node must only be advertised to direct neighbour nodes.

The described routing protocols were originally designed for wired networks and usually used within the network layer. However wireless links are subject to decreased signal strength, multi-path propagation, noise and in some cases mobility, thus routing protocols for wireless networks need direct access to the physical layer and therefore can be implemented within the link layer. Although link state routing protocols have been used in research for wireless multi-hop networks, their suitability can be disputed because they need a complete knowledge of the network at every node in order to execute a Dijkstra based graph processing algorithm. In general on demand source or distance vector protocols seem to be better suited for the dynamic nature of wireless multi-hop networks, because they work with impartial information on nodes, where only the topology of direct neighbours is known. Their general critique however is a high messaging overhead and path discovery latency compared to link state routing.

Traditionally the cost for a path is modelled as a sum of costs for each link on each hop of a path according to equation 2.5. Where n is the number of hops between source and destination and the cost can be an estimated or measured link quality such as Round Trip Time (RTT), ETX [22], ETT [29] and its variant ALM [88], or a simple hop count.

$$cost(P) = \sum_{i=1}^{n} cost_i \tag{2.5}$$

Hop count ignores a links error rate and its available bandwidth, therefore it is not suited for wireless networks. ETX, defined in equation 2.6 solves the problem by consideration of the probability p that a packet needs retransmission due to corruption, but fails to account for the available bandwidth on the wireless link.

$$ETX = \frac{1}{1-p} \tag{2.6}$$

The available bandwidth on a wireless link is dynamic and determined by factors such as distance, wavelength, transmission power, antenna gain and channel noise. On a noisy channel or a long range link the modulation scheme used is usually robust but low bit rate, compared to a noise less channel or small distance between sender and receiver. ETT, defined in equation 2.7 takes the available bandwidth into account and applies is to ETX to compute an estimated duration for a transmission. Parameter b denotes the packet size of a probe packet and r represents the bit rate of the link.

$$ETT = (\frac{b}{r}) \cdot ETX \tag{2.7}$$

Airtime Link Metric (ALM), defined in equation 2.8 basically represents ETT increased by the duration necessary to access the channel, the channel access overhead (o).

$$ALM = (o + \frac{b}{r}) \cdot ETX \tag{2.8}$$

The general goal of routing metrics is to identify a path with minimal total cost. With the introduction of MIMC to multi-hop networks in order to realize more concurrent communication on spectrum in space and time the traditional summation approach is inadequate. To maximize the benefit of multiple channels, it is necessary to use routes where constituting links cause least interference among each other. The problem with the summation approach is that links within a path are examined in isolation [23] of each other as independent entities. However links that use the same channel in vicinity of each other interfere and thus are interdependent. Therefore path cost calculation must consider interdependencies of links and discriminate between paths with different amount of interference. An innovation to the path cost calculation problem is the introduction of context aware [23] routing metrics in literature. Context means that the cost calculation is aware of technologies offered by lower layers such as multiple interfaces and non overlapping channels, link layer network coding [43] or other techniques. In our case the context is the availability of MIMC architecture and awareness of interference, therefore we refer to the routing metrics as interference aware.

Interference aware metrics are WCETT, Metric of Interference and Channel switching (MIC) [90], [91], Self Interference aware Metric (SIM) [23] , Exclusive Expected Transmission Time (EETT) [39], interference AWARE routing metric (iAWARE) [81], Cumulative Interference Metric (CIM) [10].

WCETT is defined in equation 2.9, here i denotes the i-th hop on path P, β is a weighting factor between 0 and 1. The left term is the sum of ETT along a path, the second term is the penalizing effect of channel reuse along the path. If a certain channel is reused multiple times along a path it is considered to cause IaFI.

$$WCETT(P) = (1-\beta) \cdot \sum_{i=1}^{n} ETT_i + \beta \cdot \max_{1 \leq j \leq k} X_j \tag{2.9}$$

$$X_j = \sum_{Hop\ i\ is\ on\ channel\ j} ETT_i \ 1 \leq j \leq k$$

Notice that only the channel with the highest ETT and most frequency of reuse is considered. This is due to the notion that the link with the lowest capacity restricts the overall capacity of a flow. An analogy to this phenomenon is the flow of a fluid through a conduit. No matter how large the diameter of most of the segments of a conduit is, its long term flow throughput is limited by its smallest segment. This restriction is also referred to as a bottleneck segment or link. WCETT introduces the important concept of awareness of IaFI and penalization of paths with channel uniform hops. In general it prefers paths with a balanced amount of channel uniform hops. The main critique of WCETT is that it penalizes long paths inappropriately compared to short paths if the same channel is reused. It assumes that all links within a path interfere with each other no matter how far apart they are. Though as we have discussed in section 2.1.1 the energy level of a radio signal decreases proportionally to the inverse square law of the distance between sender and receiver, thus WCETT ignores this basic fact. Therefore its performance advantage is limited to short paths as we will comprehend in our simulative experiments in section 5.2.3. WCETT can be used with link state routing such as Context-based path Pruning (CPP) [23], source routing such as MR-LQSR [29] and distance vector based routing such as HWMP as reported in literature [10],[19] and is generally used as a baseline for comparisons with more advanced metrics.

MIC, defined in equation 2.10, here $\alpha > 0$ is a weighting factor to balance the two terms. The first term Interference aware Resource Usage (IRU) is a sum of ETT of links along the path. Each links ETT is multiplied by N_l, the number of neighbour nodes that use the same channel, to penalize the path cost for IrFI. The second term Channel Switching Cost (CSC) is used to reflect IaFI, it penalizes the path cost if the previous hop uses the same channel as hop i.

$$MIC(P) = \alpha \cdot \sum_{link_l \in P} IRU_l + \sum_{node_i \in P} CSC_i \tag{2.10}$$

$$IRU_l = ETT_l \cdot N_l$$

$$CSC_i = \begin{cases} w_1 & \text{if } CH(prev(i)) \neq CH(i) \\ w_2 & \text{otherwise.} \end{cases}$$

$$0 \leq w_1 < w_2$$

The availability of N_l at a given link is a practical challenge. Also MIC is criticized for its limited memory span [23] which can lead to pathological path formation, if used with link state routing, where a path can revisit a node more than once. MIC can be used with link state and distance vector routing type protocols such as Load and Interference Balanced Routing Algorithm (LIBRA) [91].

SIM, defined in equation 2.11, where i is the $i-th$ link along path P. The first term is the sum of ETT along all links of the path, just as the traditional calculation and the first term of WCETT. The second term is the Estimated Service Interval (ESI) at the link with the lowest capacity in the absence of contending traffic (IrFI). It gauges how fast the route can transmit packets considering the bottleneck link. Here p_{ji} is a binary value that reflects whether $link_j$ interferes with $link_i$ to penalize channel reuse, thus consider IaFI.

$$SIM(P) = (1 - \beta) \cdot \sum_i ETT_i + \beta \cdot \max_i ESI(link_i | P_{i-1}) \tag{2.11}$$

$$ESI(link_i | P_{i-1}) = ETT_i + \sum_{j<i} p_{ji} \cdot ETT_j$$

SIM is used by the authors with a centralized link state routing protocol CPP. Centralized routing protocols for wireless multi-hop networks are more suitable if networks are deployed permanently and their environment is stable. However the general unpredictability of wireless systems in terms of channel characteristics and the dynamic deployability questions the suitability of link state routing protocols for multi-hop networks. Nevertheless the composition of SIM is relatively similar to

WCETT and potentially allows its use with distance vector routing protocols such as HWMP.

EETT is defined in equation 2.12, where l is the $l-th$ link of path P. $EETT_l$ is defined as the sum of ETT of all links in the Interference Set (IS) of link l, itself included. This metric penalizes both IaFI and IrFI.

$$EETT(P) = \sum_{l}^{n} EETT_l \tag{2.12}$$

$$EETT_l = \sum_{link_i \in IS(l)} ETT_i$$

Its primary design goal is to evaluate long paths with multiple available channels. EETT is used with OLSR, the authors report minor average per-flow throughput improvement compared to WCETT.

iAWARE, defined in equation 2.13, uses the same equation structure as WCETT but weighs ETT with an Interference Ratio (IR). IR is a value between 0 and 1 which describes the signal power level of a packet at the receiver in relation to physically measure interference from neighbouring nodes. For details on the calculation of IR, please refer to [81].

$$iAWARE(P) = (1-\beta) \cdot \sum_{i=1}^{n} iAWARE_i + \beta \cdot \max_{1 \le j \le k} X_j \tag{2.13}$$

$$X_j = \sum_{conflicting\ link\ i\ is\ on\ channel\ j} iAWARE_i \ 1 \le j \le k$$

$$iAWARE_i = \frac{ETT_i}{IR_i}$$

iAWARE is used by the authors with a variant of distance vector based routing protocol AODV [66]. Since the cost calculation formula the same as WCETT the metric penalizes interference of neighboring links in the same way as distant links, which impairs its optimization potential for long paths.

CIM is a WCETT based metric, which increases ETT if links within communication range of a given link use the same channel. Except for ETT it uses CIM defined in equation 2.14, where CIM for link (u, v) using channel j is Z (packet size) over Interferer-link Bit Rate (IBR). IBR is a ratio of the bit rate r_j of link u, v over the set S of neighbour links using channel j, itself included. The basic difference to WCETT is the implicit examination of IrFI in addition to IaFI.

$$CIM_j(u,v) = ETX(u,v) \cdot \frac{Z}{IBR_j(u,v)} \tag{2.14}$$

$$IBR_j(u,v) = \frac{r_j(u,v)}{|S_j(u,v)|+1}$$

CIM is used by the authors with the distance vector based routing protocol HWMP it compared to WCETT and ALM with modest improvement in packet throughput and packet loss.

The challenge of exploiting multiple channels lies not only in more sophisticated context aware cost calculation techniques compared to traditional sum calculations, but also in actually finding the path with the minimal cost according to the chosen equation. Path finding is realized by a path formation mechanism of the underlying routing protocol and is as much important at the cost calculation itself, because the most advance composite metric does not help to improve network performance if the path formation mechanism prevents the optimal route from being found. The IEEE 802.11s draft standard is designed to form WMNs, which are an aggregation of nodes in a mesh topology. In a mesh topology, participating nodes collaborate to propagate data packets from one point to another. They can be static or mobile or consist of both types of nodes. WMNs can have gateway nodes which are connected to infrastructure networks with access to Internet Services. It specifies a mandatory implementation of HWMP on routing devices in order to achieve compatibility. HWMP is a hybrid protocol with two modes a proactive mode, and a reactive mode. The proactive mode uses a tree structure, where a gateway node is the tree root. The reactive mode forms paths on demand. Both modes are based on distance vector routing. In this thesis we choose HWMPs reactive component as the path formation mechanism and show that a simple substitution of its original cost calculation metric ALM with some of the above mentioned context aware metrics cannot guarantee optimal path discovery with regard to IaFI if multiple links with non overlapping channels are available between nodes (CCA). In the following chapter we show with examples HWMPs deficient path formation mechanism.

2.3 Chapter Summary

In this chapter we have provided a background our field of interest, Wireless Multi-Hop Networks. We have shown that these networks suffer severe performance problems especially if they become large. We have identified two types of interference as the main cause and presented state of the art research that aims to remedy the interference problem by introduction of multiple channels. Then we have discussed channel assignment and routing concepts and determined CCA as assumed scheme for our system model. Finally we have discussed routing concepts in wireless multi-hop networks where traditional cost calculation falls short of considering underlying technologies such as multiple channels, which can be used to reduce interference. Hence we have examined routing metric concepts which aim at penalizing paths with links that are subject to interference in order to find optimal paths and improve network performance. In the following chapter we will define our system model formally and demonstrate the problem of this work with the help of examples, which focus is on the exploitation of channel diversity if a CCA scheme is used in combination with distance vector routing protocols.

Chapter 3

Problem Statement

Contents

3.1 System Model

In this thesis IEEE 802.11s based multi-hop networks with CCA are considered as shown in figure 3.1.

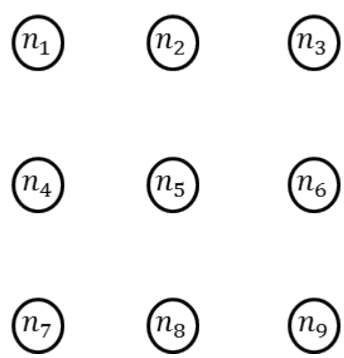


Figure 3.1: System Model

A set of n nodes i.e. $N = \{n_1, n_2, ..., n_n\}$ exists. Each node has a set of k radio interfaces i.e. $R = \{r_1, r_2, ..., r_k\}$. Each radio interface is assigned a fixed orthogonal channel from the set $C = \{c_1, c_2, ..., c_k\}$ so that $r_1^{c1} \neq r_2^{c2}, ... \neq r_m^{cm}$. The distance between nodes is chosen so that they only associate and communicate with immediate neighbours directly. Because we assume Common Channel Assignment (CCA), all channels are used for association i.e. node n_1 can reach node n_2 via r_1^{c1} or r_2^{c2} ... or r_m^{cm}, therefore the nodes are k-connected. Although CCA creates a limited amount of channel diversity compared to Varying Channel Assignment (VCA) it is more practical, due to its simplicity as discussed in section 2.2.1.5. The nodes are located at an equal distance (grid) so that communication range R_C is one hop, while interference range R_I is higher, we assume about 3 hops and determine it experimentally in section 5.2.1. The reason for grid position allocation is that the i-th hop of a path does not interfere with $i-3$-th and $i+3$-th hop and other more distant hops. This allows minimize IaFI to the highest degree. However with a relaxed allocation assumption such as random position allocation, our concept should still work, although with a reduced benefit. Every channel has a certain channel characteristic for a given pair of neighbour nodes expressed as channel metric e i.e. $e^{c_1}_{n_1 n_2}$ is the estimated metric for the link between node n_1 and n_2 on c_1 based on historical packet loss on that link, available channel bandwidth and a hypothetical probe packet of a certain size.

3.2 Distance Vector Path Formation Problem

Given our system model in section 3.1 a potentially discovered path between a source and destination node may be represented by figure 3.2. In a grid multiple such paths exist. Here we assume that the source of a packet flow is node x_1 and the destination is node x_5. Further we assume $|S| = 3$ and for simplicity let $c_1 = A$, $c_2 = B$ and $c_3 = C$ as depicted in figure 3.2. The estimated channel metrics are expressed with exemplary values at every link. For example $e^{A}_{SI_1} = 100$ or $e^{C}_{I_3 D} = 200$.

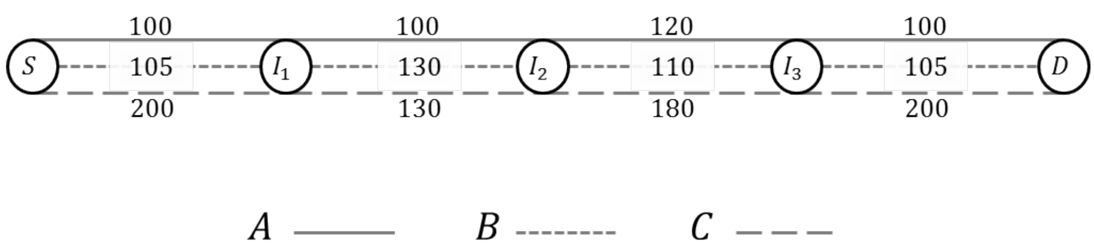


Figure 3.2: Example Path MIMC

For a better understanding of the problem we illustrate how HWMP and its metric ALM establish a path as a baseline metric example. This reveals two basic problems HWMP encounters when a MIMC architecture in combination with CCA is present. The first problem is its cost calculation if used with ALM which is based on summation as discussed in section 2.2.2. The second problem is Forward Path Unawareness (FPU) if used with any routing metric, a term which will be defined in the sections to follow. The second problem is less severe if VCA is assumed, however it exists if more than one common channel between two nodes are available between two nodes. Then two other examples are given to demonstrate how the IaFI interference aware metrics WCETT and SIM as representatives for metric based concepts and our proposed metric Cumulative Interference aware Expected Transmission Time (CIETT) try to solve one of the problems.

Because multiple channels between the nodes are available the path setup of HWMP depends on link metrics. Node S broadcasts a PREQ frame on all channels in consecutive order A, B, C with very short time in between. When node I_1 receives the first PREQ frame on channel A, the cumulative ALM is 100. It then creates a reverse path entry to node S and broadcasts the PREQ on all channels. When node I_1 receives the second PREQ frame with the same sequence number from node S on channel B, it compares its metric 105 with the metric of its existing reverse path entry 100. Because the existing reverse path has a better link metric, it ignores the PREQ frame from channel B. The same happens to the PREQ frame from node S on channel C. When node I_2 receives the first PREQ frame from node I_1 on channel A, it creates a reverse route entry to node S via node I_1 and broadcasts the PREQ on all channels. The PREQ frames form node I_1 on channel B and C are ignored because their cumulative ALM is higher than the known metric for the reverse path to node S. In this case $(e^A_{SI_1} + e^A_{I_1I_2} = 100 + 100 = 200) \leq (e^A_{SI_1} + e^B_{I_1I_2} = 100 + 130 = 230) \leq (e^A_{SI_1} + e^C_{I_1I_2} = 100 + 130 = 230)$. Table 3.1 demonstrates the path setup at every node.

Table 3.1: Example HWMP with ALM

Time	Node	PREQ	Metric Calculation	ALM	Status
t_a		$PREQ^{A}_{SI_1}$	0+100	100	accepted
t_{a+1}	I_1	$PREQ^{B}_{SI_1}$	0+105	105	ignored
t_{a+2}		$PREQ^{C}_{SI_1}$	0+200	200	ignored
t_b		$PREQ^{AA}_{SI_1;I_1I_2}$	100+100	200	accepted
t_{b+1}	I_2	$PREQ^{AB}_{SI_1;I_1I_2}$	100+130	230	ignored
t_{b+2}		$PREQ^{AC}_{SI_1;I_1I_2}$	100+130	230	ignored
t_c		$PREQ^{AAA}_{SI_1;I_1I_2;I_2I_3}$	100+100+120	320	overwritten by AAB
t_{c+1}	I_3	$PREQ^{AAB}_{SI_1;I_1I_2;I_2I_3}$	105+130+110	310	accepted
t_{c+2}		$PREQ^{AAC}_{SI_1;I_1I_2;I_2I_3}$	200+130+180	380	ignored
t_d		$PREQ^{AAAA}_{SI_1;I_1I_2;I_2I_3;I_3D}$	100+100+120+100	420	overwritten by $AABA$
t_{d+1}		$PREQ^{AAAB}_{SI_1;I_1I_2;I_2I_3;I_3D}$	100+100+120+105	425	ignored
t_{d+2}	D	$PREQ^{AAAC}_{SI_1;I_1I_2;I_2I_3;I_3D}$	100+100+120+200	520	ignored
t_{d+3}		$PREQ^{AABA}_{SI_1;I_1I_2;I_2I_3;I_3D}$	100+100+110+100	410	accepted
t_{d+4}		$PREQ^{AABB}_{SI_1;I_1I_2;I_2I_3;I_3D}$	100+100+110+105	415	ignored
t_{d+5}		$PREQ^{AABC}_{SI_1;I_1I_2;I_2I_3;I_3D}$	105+130+110+200	510	ignored

Notice that the PREQ frames are received at different points in time a, b, c, d with $a < b < c < d$. The notion t_{a+1} means that this PREQ was received after a previous PREQ at t_a. This is due to the fact that the implementation processes a PREQ in sequential order. When the PREQ needs to be broadcasted, it is first sent on the first interface, then the second and so on. How much later the same PREQ frame is received at a node on different channels is determined by the processing, medium access and transmission delay on the channels. The time of arrival is important for the acceptance check and therefore for the path formation and final channel sequence combination. As a result HWMP with ALM sets up the following path S^A, I_1^A, I_2^B, I_3^A. According to the system model, hop S^A interferes with I_1^A, as well as I_1^A with I_3^A. The effect is double for I_1^A because it interferes with two hops. Hence IaFI exists on 3 out of 4 hops along the path.

HWMP with ALM shows two problems in above situation. First it employs a simple cumulation of estimated link metrics and therefore ignores available orthogonal channels. Thus it cannot prefer link combinations with less channel uniform hops. This reduces network performance because every consecutive channel uniform hop has to share its bandwidth with its neighbour links within R_I. The second problem is that intermediate nodes, when faced with the decision if a PREQ frame with the same or higher metric should be accepted, ignore those frames. This is necessary because only one (reverse) path entry must exist. However this leads to the exclusion of links such as SI_1^B or SI_1^C. Because PREQ frames are broadcasted intermediate nodes have no knowledge along which node on which channel with which metric the frame will travel. Since this information is not available, intermediate nodes are unaware to the right side of the path to the destination, the forward path, we term this lack of knowledge Forward Path Unawareness (FPU). This mechanism results in a locally optimal decision for intermediate nodes, but because it excludes some possible links globally (for the whole path) the channel sequence combination is not necessarily optimal.

In literature several concepts have been proposed to solve the first problem of channel diversity ignorance with WCETT, CIM, SIM and other context aware metric based approaches. To understand how these concepts improve the path discovery compared to ALM we give the following example using HWMP with WCETT in table 3.2. We apply a β of 0.5, the metric is defined by equation 2.9 in section 2.2.1.4.

Consequently HWMP with WCETT finds and establishes a path with the following channel sequence A, A, B, B with a metric of 315. It is clear that WCETT offers better use of channel diversity while preferring links with lower metric values. However links SI_1; I_1I_2 and I_2I_3; I_3D still interfere with each other under our system model. WCETT tries to balance channel uniform hops along a path as much as possible. Its general critique is that it does not consider the distance between channel uniform hops. Further it still suffers from FPU and excludes e.g. $PREQ^B_{SI_1}$ or $PREQ^{AC}_{SI_1;I_1I_2}$.

The WCETT related metric SIM as defined in equation 2.11 assumes a more refined interference model, where two links interfere with each other if one of the nodes of the two links is within Communication Range (R_C) of one other node of the two links according to [23]. However we argue that Interference Range (R_I) should be used instead, since a link can interfere with another link on a much greater distance than

Table 3.2: Example HWMP with WCETT

Time	Node	PREQ	Metric Calculation	WCETT	Status
t_a		$PREQ^{A}_{SI_1}$	(100+100)/2	100	accepted
t_{a+1}	I_1	$PREQ^{B}_{SI_1}$	(105+105)/2	105	ignored
t_{a+2}		$PREQ^{C}_{SI_1}$	(200+200)/2	200	ignored
t_b		$PREQ^{AA}_{SI_1;I_1I_2}$	(200+200)/2	200	overwritten by AB
t_{b+1}	I_2	$PREQ^{AB}_{SI_1;I_1I_2}$	(230+130)/2	180	accepted
t_{b+2}		$PREQ^{AC}_{SI_1;I_1I_2}$	(230+130)/2	180	ignored
t_c		$PREQ^{AAA}_{SI_1;I_1I_2;I_2I_3}$	(320+320)/2	320	overwritten by AAB
t_{c+1}	I_3	$PREQ^{AAB}_{SI_1;I_1I_2;I_2I_3}$	(310+200)/2	255	accepted
t_{c+2}		$PREQ^{AAC}_{SI_1;I_1I_2;I_2I_3}$	(380+200)/2	290	ignored
t_{c+3}		$PREQ^{ABA}_{SI_1;I_1I_2;I_2I_3}$	(350+220)/2	285	ignored
t_{c+4}		$PREQ^{ABB}_{SI_1;I_1I_2;I_2I_3}$	(340+240)/2	290	ignored
t_{c+5}		$PREQ^{ABC}_{SI_1;I_1I_2;I_2I_3}$	(410+180)/2	295	ignored
t_d		$PREQ^{AAAA}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(420+420)/2	420	overwritten by $AAAB$
t_{d+1}		$PREQ^{AAAB}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(425+320)/2	373	overwritten by $AABA$
t_{d+2}	D	$PREQ^{AAAC}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(515+320)/2	418	ignored
t_{d+3}		$PREQ^{AABA}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(410+300)/2	355	overwritten by $AABB$
t_{d+4}		$PREQ^{AABB}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(415+215)/2	315	accepted
t_{d+5}		$PREQ^{AABC}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(505+200)/2	353	ignored

R_C. Thus we use our interference model of three hops for the calculation of cost with SIM. The general idea of SIM is to penalize a paths cost if links interfere with each other along the path in form of IaFI. It considers the bottle neck link of a path while WCETT considers the bottleneck channel. Table 3.3 presents the path formation steps of HWMP in combination with SIM with $\beta = 0.5$.

As one can see SIMs cost calculation is very similar to WCETT in most cases, however it results in the channel sequence $AABA$ in our example, same as interference unaware ALM, which is IaFI prone. It must be noted that SIM is primarily designed for a centralized link state routing protocol but is generally considered suitable for distance vector routing.

For further analysis of approaches that substitute interference blind metrics by interference aware metrics with HWMP, we design CIETT. For the computation of CIETT, a route record is collected along the path on which a PREQ frame travels from source to destination. This route record contains two previous links, their channel numbers and ETT values. The metric is calculated as specified in equation 3.1.

$$CIETT(P) = \sum_{i=1}^{n} (ETT_i + a_i(ETT_{i-1}) + b_i(ETT_{i-2})) \qquad (3.1)$$

Table 3.3: Example HWMP with SIM

Time	Node	PREQ	Metric Calculation	SIM	Status
t_a		$PREQ^{A}_{SI_1}$	(100+100)/2	100	accepted
t_{a+1}	I_1	$PREQ^{B}_{SI_1}$	(105+105)/2	105	ignored
t_{a+2}		$PREQ^{C}_{SI_1}$	(200+200)/2	200	ignored
t_b		$PREQ^{AA}_{SI_1;I_1I_2}$	(200+200)/2	200	overwritten by AB
t_{b+1}	I_2	$PREQ^{AB}_{SI_1;I_1I_2}$	(230+130)/2	180	accepted
t_{b+2}		$PREQ^{AC}_{SI_1;I_1I_2}$	(230+130)/2	180	ignored
t_c		$PREQ^{AAA}_{SI_1;I_1I_2;I_2I_3}$	(320+320)/2	320	overwritten by AAB
t_{c+1}	I_3	$PREQ^{AAB}_{SI_1;I_1I_2;I_2I_3}$	(310+200)/2	255	accepted
t_{c+2}		$PREQ^{AAC}_{SI_1;I_1I_2;I_2I_3}$	(380+200)/2	290	ignored
t_{c+3}		$PREQ^{ABA}_{SI_1;I_1I_2;I_2I_3}$	(350+220)/2	285	ignored
t_{c+4}		$PREQ^{ABB}_{SI_1;I_1I_2;I_2I_3}$	(340+240)/2	290	ignored
t_{c+5}		$PREQ^{ABC}_{SI_1;I_1I_2;I_2I_3}$	(410+180)/2	295	ignored
t_d		$PREQ^{AAAA}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(420+420)/2	420	overwritten by $AAAB$
t_{d+1}		$PREQ^{AAAB}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(425+320)/2	373	overwritten by $AABA$
t_{d+2}	D	$PREQ^{AAAC}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(515+320)/2	418	ignored
t_{d+3}		$PREQ^{AABA}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(410+200)/2	305	accepted
t_{d+4}		$PREQ^{AABB}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(415+215)/2	315	ignored
t_{d+5}		$PREQ^{AABC}_{SI_1;I_1I_2;I_2I_3;I_3D}$	(505+200)/2	353	ignored

$$a_i = \begin{cases} 1 & \text{if } ch_i = ch_{i-1} \\ 0 & \text{otherwise} \end{cases} \quad b_i = \begin{cases} 1 & \text{if } ch_i = ch_{i-2} \\ 0 & \text{otherwise} \end{cases}$$

The Boolean variables a and b are true if one of the two previous channels is reused on hop_i, else false. The metric is larger for paths with channel uniform hops within interference range of three hops.

The metric for a path P is the sum of all ETT values of the links the PREQ frame has travelled, plus the ETT of the previous hop if the same channel has been used as the current hop, plus the ETT of the pre-previous hop if the same channel has been used as the current hop. Consequently CIETT increases the metric for channel uniform hops within a two hop distance of the current hop. This has the effect that PREQ frames that travelled a channel diverse path tend to have a lower metric value compared to ones that were transmitted on channel uniform paths, which results in preference, acceptance, reverse routing table entry/update and rebroadcast.

Table 3.4 shows that HWMP with CIETT forms a path with the channel sequence $AABC$. Although this combination is more channel diverse than obtained with ALM ($AABA$) and WCETT ($AABB$) it still fails to establish a complete channel diverse combination. It seems, that the effect of Forward Path Unawareness is too strong to be mitigated by an interference aware routing metric.

We show in section 4.4 that if R_I is 3 hops the best channel combination with

Table 3.4: Example HWMP with CIETT

Time	Node	PREQ	Metric Calculation	CIETT	Status
t_a	I_1	$PREQ^{A}_{SI_1}$	0 + 100	100	accepted
t_{a+1}		$PREQ^{B}_{SI_1}$	0 + 105	105	ignored
t_{a+2}		$PREQ^{C}_{SI_1}$	0 + 200	200	ignored
t_b	I_2	$PREQ^{AA}_{SI_1;I_1I_2}$	100+100+100	300	overwritten by AB
t_{b+1}		$PREQ^{AB}_{SI_1;I_1I_2}$	100+130	230	accepted
t_{b+2}		$PREQ^{AC}_{SI_1;I_1I_2}$	100+130	230	ignored
t_c	I_3	$PREQ^{AAA}_{SI_1;I_1I_2;I_2I_3}$	300+120+100+100	620	overwritten by AAB
t_{c+1}		$PREQ^{AAB}_{SI_1;I_1I_2;I_2I_3}$	300+110	410	accepted
t_{c+2}		$PREQ^{AAC}_{SI_1;I_1I_2;I_2I_3}$	300+180	480	ignored
t_{c+3}		$PREQ^{ABA}_{SI_1;I_1I_2;I_2I_3}$	230+120+100	450	ignored
t_{c+4}		$PREQ^{ABB}_{SI_1;I_1I_2;I_2I_3}$	230+110+130	470	ignored
t_{c+5}		$PREQ^{ABC}_{SI_1;I_1I_2;I_2I_3}$	230+180	410	ignored
t_d	D	$PREQ^{AAAA}_{SI_1;I_1I_2;I_2I_3;I_3D}$	620+100+120+100	940	overwritten by $AAAB$
t_{d+1}		$PREQ^{AAAB}_{SI_1;I_1I_2;I_2I_3;I_3D}$	620+105	725	overwritten by $AABA$
t_{d+2}		$PREQ^{AAAC}_{SI_1;I_1I_2;I_2I_3;I_3D}$	620+200	820	ignored
t_{d+3}		$PREQ^{AABA}_{SI_1;I_1I_2;I_2I_3;I_3D}$	410+100+100	610	overwritten by $AABC$
t_{d+4}		$PREQ^{AABB}_{SI_1;I_1I_2;I_2I_3;I_3D}$	410+105+110	625	ignored
t_{d+5}		$PREQ^{AABC}_{SI_1;I_1I_2;I_2I_3;I_3D}$	410+200	610	accepted

minimal IaFI is A, C, B, A. From our examples one can see that this combination is prevented by the route setup mechanism of HWMP or in general AODV. Because node x_3 has the decision to accept or ignore a PREQ frame with the same or higher metric value, but does not know which metric values will follow on links to the right of the path, it excludes channel C at hop I_1I_2 from being considered for route setup. We argue that even with more sophisticated metric calculations, which are aware of channel diversity, the established paths will be suboptimal because not all channel combinations along a path are considered.

3.3 The Problem and Hypothesis

The core of our the problem *is therefore the fact that optimization decisions with AODV based routing protocols are made at intermediate nodes, which are subject to Forward Path Unawareness and thus exclude potential channel sequences which may lead to optimal results with respect to minimal IaFI on k-connected topologies that apply Common Channel Assignment.*

To solve this problem we propose to forgo the distributed optimization approach on intermediary nodes and relocate the optimization to the destination node. At the destination node the path terminates, hence cannot be subject to Forward Path Unawareness. To realize this relocation it is necessary to collect information about all available links at every hop of the path. We introduce the concept of a route record to HWMP inspired by DSR. The route record shall be used purely for distance vector path optimization and establishment, not for operation contrary to DSR.

Our Hypothesis *is that Route record based Distance Vector (RDV) can find and establish a path with less IaFI and thus improve network performance by optimizing the channel sequence along a path compared to HWMP with an interference aware metric.*

Minimal IaFI means less channel contention, less noise, higher SINR and increase in data throughput, packet delivery fraction and less packet delay.

3.4 Chapter Summary

In this chapter we have provided the reader a statement of the problem of distance vector based routing protocols in particular HWMP with examples that show their inability to find and setup paths with a channel sequence combination with minimal IaFI. Chapter 4 describes our concept of RDV, presents a channel sequence selection problem, a theoretical analysis and potential solutions. Chapter 5 presents evidence from experimentation with regard to our hypothesis.

CHAPTER 4

RDV: Route Record Distance Vector

Contents

In this chapter we provide a description of our proposed protocol, state the optimization problem, give a theoretical analysis and present how the Viterbi algorithm based on a second order Markov property can solve the problem. For efficiency reasons we propose an approximative greedy algorithm inspired by Viterbi.

4.1 Path Formation Protocol Mechanism

The principle of RDV can be explained by comparing it to the principle of AODV and also HWMP path formation. Figure 4.1 shows two phases. We assume that the network state is reset i.e. no routes exist. Before any routes can be setup, nodes must associate with each other on their available channels. Peer Management Protocol (PMP) is used for association of two nodes on a common channel within R_C. Nodes periodically broadcast beacon frames which advertise their participation in a 802.11s network. When a current node receives such a frame from a neighbour node, it reacts by transmission of an association request in form of a *peerLinkOpen* frame. The neighbour node then replies with a $peerLinkConfirm$ frame. The neighbour node repeats the procedure with a *peerLlinkOpen* frame which is replied by the current node with a $peerLinkConfirm$ frame. This procedure is termed four-way hand shake. The *peerLinkClose* frame is sent if:

- a node leaves the Basic Service Set (BSS)Basic Service Set,
- a link fails due to unspecified reasons,
- the PMP finite state machine holding timer has expired.

Notice that if a packet cannot be transmitted due to contention on a link several times, this is assumed a link failure and it triggers a *peerLinkClose* frame. After peer links are setup the path formation mechanism is activated at the source node on demand, if a higher layer application needs to communicate with a destination node. On the left side the broadcast (dotted circle) phase is depicted, during which node S broadcasts a PREQ frame with node D as destination. Every intermediate node that receives the frame, creates a temporary reverse path entry in its routing table to node S and rebroadcasts the PREQ frame. If an intermediate node receives a PREQ frame with the same sequence number, it ignores it unless it carries a better path metric. This behaviour gives intermediate nodes optimization control. If node D receives a PREQ frame, it creates a reverse path entry to node S.

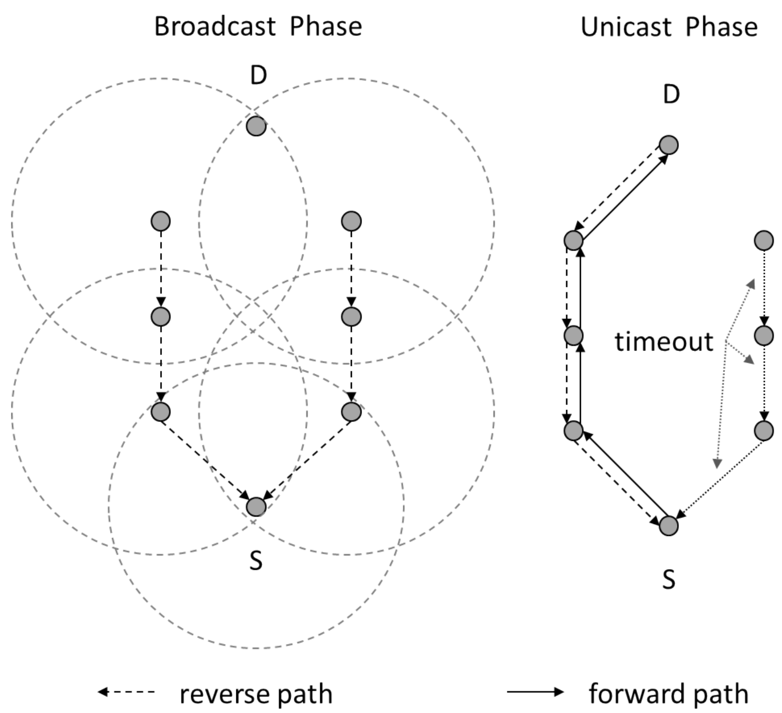


Figure 4.1: AODV/HWMP Path Formation

On the right side, the unicast phase is depicted, during which node D generates a PREP frame and sends it to node S using its reverse path entry. The shown topology allows two possible routes from S to D, using the left hand intermediate nodes or the right hand ones. Every PREQ frame carries a routing metric field, which stores the cumulative path metric of all links visited by the frame. If node D receives the PREQ frame along the left hand route first, it replies using this route. If the PREQ frame from the right hand route has an equal or higher path metric it is ignored, which we assume in this example, otherwise it will reset the route and answer with a PREP frame. When intermediate nodes receive a PREP frame from node D, they create forward path routing table entries to node D and mark the temporary reverse route entry to S as permanent (for their lifetime). Node S does the same, except for forwarding. Temporary path entries on the right hand side time out and are deleted.

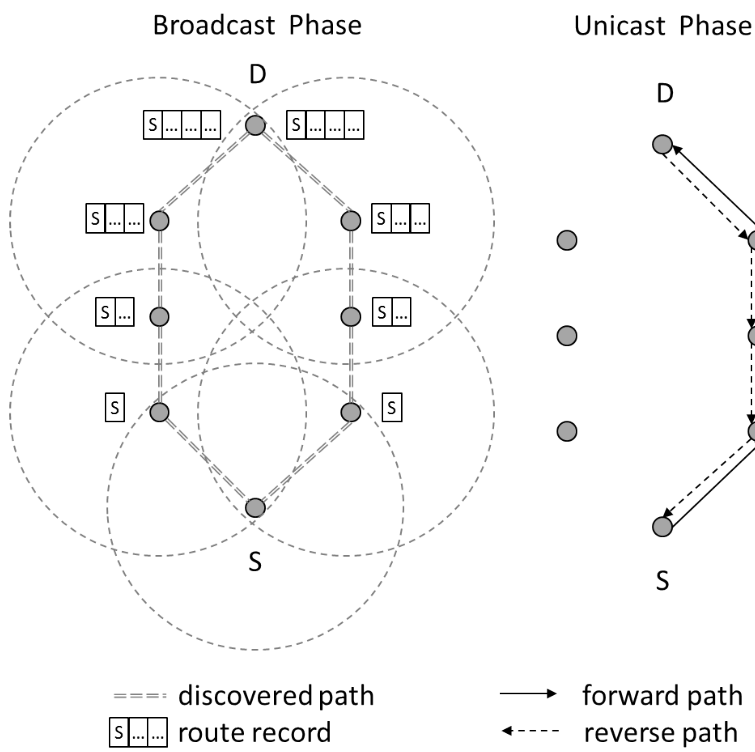


Figure 4.2: RDV Path Formation

Figure 4.2 displays the path formation procedure of RDV. It uses the same concepts as AODV/HWMP, mainly its sequence number mechanism to prevent cyclic paths and the same broadcast (PREQ) and unicast (PREP) phases. On the left side of the figure the broadcast phase is presented. When necessary node S broadcasts a PREQ frame with node D as destination. Node S adds its route record to the frame, which contains its identity, all its available interface addresses, their channel numbers and an empty metric field for each channel. If an intermediate node receives the frame it does not create a reverse path entry to node S. It measures the metric for all of S channels and updates the route record of S. Further it adds a route record to the frame for itself and broadcasts it. On the right side of the figure the unicast phase is displayed. If node D receives a PREQ frame, it reads its route record, finds the channel sequence with the best path metric, creates a reverse route entry to node S and sends a PREP frame using its reverse path routing table entry. The PREP frame contains the complete route record from S to D. If an intermediate node receives the PREP frame, it reads the route record and creates both reverse path to S and forward path to D in its routing table. It forwards the PREP frame to S using its reverse path. Node S does the same as intermediate nodes except for forwarding. As AODV/HWMP, RDV will ignore a PREQ frame with the same sequence number and equal or higher path metric, otherwise it will accept, reset the reverse route and reply. We assume here that the right hand path has a better metric, because RDV determined that the channel sequence combination is better than on the left hand route.

Following we present the PREQ/PREP frame structure of HWMP in comparison to RDV. Figure 4.3 shows a simplified representation with field names and their size measured in octets. It contains an Originator (source) Address and Destination Address and a field for a cumulative metric. It omits sequence number fields and various protocol flags. For further detail please refer to [88].

Originator Address	Destination Address	Metric
6 octets	6 octets	4 octets

Figure 4.3: PREQ/PREP Frame Structure of HWMP

Figure 4.4 displays the modified frame structure of RDV. Besides an Originator and Destination Address, the metric field is also present, but unused. Instead a route record is appended at the end of the frame, which contains one to n entries, with n being the length of the path travelled by the PREQ frame between source and destination node.

Originator Address	Destination Address	Metric (unused)	Route Record
6 octets	6 octets	4 octets	variable length

Figure 4.4: PREQ/PREP Frame Structure of RDV

A route record entry is added by the source node and each intermediate node. Figure 4.5 shows the details of a route record entry. Each entry contains the main address of the node, which is used as a unique identifier, generally this address is the Medium Access Control (MAC) Address of the first radio interface. It further contains one to m interface entries, m being the number of installed interfaces. Each interface entry contains information about interface MAC address, channel number, channel metric, channel selection flag and IaFI penalty factor.

Nodes Main Address 6 octets	Iface 1 Address 6 octets	Iface 1 Channel 1 octet	Iface 1 Metric 4 octets	Iface 1 Select.Flag 1 octet	Iface 1 IaFI Penalty 2 octets	...	Iface m Address 6 octets	Iface m Channel 1 octet	Ifacem Metric 4 octets	Iface m Select.Flag 1 octet	Iface m IaFI Penalty 2 octets

Figure 4.5: Structure of a Route Record Entry of RDV

Notice that link metric for each interface is calculated by the node that receives the PREQ frame according to equation 2.8 and then written into the route record. The metric for each link of a node that transmitted the PREQ frame is calculated as follows. When an node receives a PREQ frame, it reads the MAC address of the first interface of m interfaces of the PREQ sender from the route record and determines the amount of channel resources that would be consumed if a frame of a certain size is successfully transmitted over that particular link to the given MAC address. The calculation depends on the current link bit rate and frame error rate. IEEE 802.11s specification leaves the estimation of the frame error rate open to the choice of implementation. In the Network Simulator 3 (NS-3) implementation [4] of HWMP this is realized as the average frame error rate for a given peer MAC-address on a given interface and channel. Every node keeps track of the success of frame transmission events to every neighbour-link and maintains a memory of average frame error rate per link. It is defined as a moving percentage of failed frames to a certain receiving interface. Every time a frame is successfully transmitted to given MAC-peer-address, the error rate is decreased, otherwise increased. The term 'moving' means the impact of a failed frame transmission on the average error rate depends on how long ago the last transmission event was. A failed transmission event will have a greater impact on the average if the last transmission event was long ago, and a smaller impact if it was recent. This calculation behaviour smooths out frame errors on highly active links compared to sparsely used links. It is important to note that no probing frames are sent for the estimation of link metrics, thus no additional overhead is created. How the average frame error rate is derived in other environments depends on the implementation and is out of scope of this work. The node that receives the PREQ frame repeats the process for all remaining $m - 1$ links.

Also notice that the link metric, which depends on a links average frame error rate can reflect interference already. If flows interfere on a certain link, they collide and produce frame errors which increases the frame error rate, which in turn increases the link metric. A viable question therefore is, why metric designs penalize (increase cost) link metric calculation if interference is already reflected. We argue, that Inter Flow Interference (IrFI) can be reflected in a link metric, if the neighbouring flow is active at the time of path discovery. But IaFI cannot be reflected, because at time of path discovery no traffic within the path exists which could create IaFI, create collisions, increase frame error rate and increase the link metric. Also notice the medium access protocol of 802.11 is designed to avoid collisions through a distributed channel sensing and medium back off mechanism, this minimizes the frame error rate but increases the contention time for neighbour nodes, so that the frame error rate reflects interference inappropriately.

The values for channel selection flag and IaFI penalty factor are zero and remain so until the PREQ frame is received and processed by node D. In above example the route record in the PREQ frame has a length of 4 entries when it arrives at node D. After node D finds the best channel sequence between itself and node S, it sets the channel selection flags. Every route record entry can only have one selected interface and channel. The IaFI penalty factor field is used for the calculation of the path metric. It describes with how many other hops the selected channel on the relevant hop has to share its spectrum in R_I. Factor p from equation 4.1 represents the IaFI penalty factor, we will elaborate this detail further in section 4.2. The PREP frame contains the route record with selected channels. Intermediate nodes use it during the PREP phase to create forward and reverse path routing table entries to node S and D. Figure 3.2 shows a four hop path. Lets assume that the selected channel sequence is $ABCA$ and the PREP frame is currently processed by node I_2. It contains the following route record entries depicted in figure 4.6, where four entries for each hop between node S and D contain routing information necessary to create distance vector routing entries.

Hop 1 Nodes Main Address S^A	$Iface_1$ Address S^A	$Iface_1$ Channel A	$Iface_1$ Metric 100	$Iface_1$ Select.Flag $True$	$Iface_1$ IaFI Penalty 0
	$Iface_2$ Address S^B	$Iface_2$ Channel B	$Iface_2$ Metric 105	$Iface_2$ Select.Flag $False$	$Iface_2$ IaFI Penalty –
	$Iface_3$ Address S^C	$Iface_3$ Channel C	$Iface_3$ Metric 200	$Iface_3$ Select.Flag $False$	$Iface_3$ IaFI Penalty –

Hop 3 Nodes Main Address I_2^A	$Iface_1$ Address I_2^A	$Iface_1$ Channel A	$Iface_1$ Metric 120	$Iface_1$ Select.Flag $False$	$Iface_1$ IaFI Penalty –
	$Iface_2$ Address I_2^B	$Iface_2$ Channel B	$Iface_2$ Metric 110	$Iface_2$ Select.Flag $False$	$Iface_2$ IaFI Penalty –
	$Iface_3$ Address I_2^C	$Iface_3$ Channel C	$Iface_3$ Metric 180	$Iface_3$ Select.Flag $True$	$Iface_3$ IaFI Penalty 0

Hop 2 Nodes Main Address I_1^A	$Iface_1$ Address I_1^A	$Iface_1$ Channel A	$Iface_1$ Metric 100	$Iface_1$ Select.Flag $False$	$Iface_1$ IaFI Penalty –
	$Iface_2$ Address I_1^B	$Iface_2$ Channel B	$Iface_2$ Metric 130	$Iface_2$ Select.Flag $True$	$Iface_2$ IaFI Penalty 0
	$Iface_3$ Address I_1^C	$Iface_3$ Channel C	$Iface_3$ Metric 130	$Iface_3$ Select.Flag $False$	$Iface_3$ IaFI Penalty –

Hop 4 Nodes Main Address I_3^A	$Iface_1$ Address I_3^A	$Iface_1$ Channel A	$Iface_1$ Metric 100	$Iface_1$ Select.Flag $True$	$Iface_1$ IaFI Penalty 0
	$Iface_2$ Address I_3^B	$Iface_2$ Channel B	$Iface_2$ Metric 105	$Iface_2$ Select.Flag $False$	$Iface_2$ IaFI Penalty –
	$Iface_3$ Address I_3^C	$Iface_3$ Channel C	$Iface_3$ Metric 200	$Iface_3$ Select.Flag $False$	$Iface_3$ IaFI Penalty –

Figure 4.6: Route Record Entries of PREP Frame

Node I_2s routing table contains following entries as presented in table 4.1:

Table 4.1: Routing Table of node I_2

Node Address	Retransmitter Address	Channel	Metric
D	I_3^C	C	280
D	I_3^C	C	180
S	I_1^B	B	230
S	I_1^B	B	130

For instance it adds a route to node D (main address), which can be reached over retransmitter I_3 on channel C. The term I_3^C represents the MAC address of node I_3s third radio interface tuned to channel C. Node I_2 knows that channel C must be used because its Selection Flag has been set to 'true' by node D. The computation of the route metric in this case is a simple summation of the channel metrics from node I_2 to nodes S and D. For instance $280 = e^C_{I_2 I_3} + e^A_{I_3 D}$. However the assumed sequence in this example $ABCA$ is channel diverse within three hops, but for a channel sequences with channel uniform consecutive hops the calculation must consider IaFI and penalize it similar to how context aware routing metrics penalize channel reuse on consecutive hops by increasing the path metric. The details of this problem are tackled in the following section 4.2.

To summarize the path formation mechanism of RDV the basic alteration compared to AODV/HWMP is the collection of a route record, complete shift of the path formation to the unicast (PREP) phase and relocation of the optimization decision from intermediate nodes to the destination node. In a single channel environment, this alteration is trivial and should not result in any modified behaviour. In fact our observation is that both routing protocols perform similar in that case. Notice, because a PREQ frame collects channel information for all available interfaces, RDV rebroadcast the frame on one random channel, instead of all channels. This reduces the broadcast overhead as will be demonstrated in section 5.2.4.

4.2 Channel Sequence Selection Problem

In this section we describe the optimization problem at the destination node. Figure 3.2 shows an example of a path discovered by a PREQ frame. Node S is the source of the PREQ frame, node D the destination, I_1, I_2, I_3 are intermediate nodes. The PREQ frame has collected a record for the route with all available channels between the nodes and their respective metric values, this information is available at node D. Our metric calculation is based on ALM, defined in equation 2.8. We use ALM to compute ETT for the implementation of WCETT and CIETT. The main motivation behind the design of our algorithm is to minimize IaFI to improve the network performance. We are interested in modifying the path metric calculation such that the amount of IaFI is reflected in it. A trivial way to minimize IaFI is simply to choose the available channels in turn on the path that results in a maximum distance between links which use the same channel. For example the channel sequence $ABCA$ would is such a channel diverse combination, where the first hop does not interfere with the fourth hop. We call this simple channel sequence algorithm *channel diverse* (or *Diverse* for simplicity) channel selection algorithm.

Although, this would help to maximize channel diversity in the channel combination, it clearly can not guarantee a high performance because the metric values play no role in the channel selection. Our goal is to combine such an idea with the metric calculation of the routing protocol to optimize the channel combination considering both channel diversity and cost minimization. One point which needs to be considered is that channel reuse influences the communication performance only if it happens in the interference range. Therefore, we need to design the metric computation function such that any channel uniformity (channel reuse) increases the cost of the path if the links with the reused channel are in each others interference range.

Based on above facts, we design the path metric, which we term Intra-flow interference aware Airtime Link Metric (IALM) as defined in equation 4.1:

$$IALM(P)_c = \sum_{i=1}^{n}(ALM_i + ALM_i \cdot p_i) \tag{4.1}$$

$$p_i = \#\{j|ch_j = ch_i\},\ for\, j \in \{i-2, i-1, i+1, i+2\}$$

Where the path metric IALM(P) for a given channel combination c, is the sum of the measured ALM values on the links on hop i to n, plus ALM_i for every channel uniform hop within a two hop distance. The penalty factor p_i is the cardinality # of the set j, which is basically a counter of all links within ± 2 hop distance of hop i. For example a channel sequence from figure 3.2 of channel A, B, A, C has the metric value $m_c^{ABAC} = 770$. The channel combination $m_c^{AAAA} = 1480$. This calculation puts emphasis on channel diversity within two hops. We must note that this calculation combined with RDV considers both directions the reverse direction and the forward direction of the path contrary to HWMP, which only allows consideration of the reverse direction, therefore this approach does not suffer from FPU and allows to consider all available links along apart. It penalizes channel uniform hops, since the reuse of the same channel within interference range results in a sharing of the available bandwidth and a sharp demise in possible throughput and other network performance metrics. The amount of possible channel sequence combinations is $|S|^n$, where $|S|$ is the number of orthogonal channels available at every hop from the set S, in this case 3 and n is the number of hops within the path. In this four hop example example the amount is $3^4 = 81$. For a 10 hop path it is 59.049 and for a 20 hop path 3.486.784.401. The naive algorithm therefore belongs to the complexity class of $O(|S|^n)$, which means that its processing demand increases exponentially and that it cannot efficiently solve the problem even for relatively short paths.

The optimization challenge is therefore finding the channel sequence combination c^* with the lowest path metric $IALM(P)_{c^*}$ from all possible sequences in a reasonable time to ensure protocol functionality.

4.3 Theoretical Analysis

The problem of channel sequence selection refers to a classical graph labelling problem. It is a *sequence labelling problem* in which all the coming graphs have the constraint to be in a chain form.

The sequence labelling problem can be generally described as follows: for a given input sequence (x_i) of tokens, we want to assign a discrete label (y_i) with y_i from certain finite alphabet S for each token in the sequence. Labels of tokens are dependent on the labels of other tokens in the sequence, particularly their neighbours.

Furthermore, as we assumed, if the hops in the route are far enough from each other, the influence between them can be ignored. This property can be used for the reduction of complexity. From a theoretical point of view this specific kind of "memorylessness" could be approximately represented by the *Markov property* which is introduced in [12].

A sequence of random variables $X_1, X_2, X_3, \ldots$ is called a *Markov chain*, if it satisfies the Markov assumption, which is when the next state only depends on the current state and is independent of the previous history. This is mathematically defined by Equation (4.2).

$$\begin{aligned} Pr(X_{n+1} = y_{n+1} | X_1 = y_1, X_2 = y_2, \ldots, \\ X_n = y_n) = Pr(X_{n+1} = y_{n+1} | X_n = y_n) \end{aligned} \tag{4.2}$$

Sometimes a Markov chain does not just have one state memory, that is to say, the future state depends on the past m states, where m is finite. In this case we have a Markov chain of order m, which is a statistical process satisfying Equation (4.3).

$$\begin{aligned} &Pr(X_n = y_n | X_1 = y_1, X_2 = y_2, \ldots, X_{n-1} = y_{n-1}) \\ &\quad = Pr(X_n = y_n | X_{n-1} = y_{n-1}, \ldots, X_{n-m} = y_{n-m}) \end{aligned} \tag{4.3}$$

Dynamic programming can be used to exploit the Markov assumption and efficiently determine the most likely state sequence for a given observation and model [36]. It can solve a complex problem by breaking it down into simpler sub problems. A problem is considered complex if the number of sub problems grows exponentially with the linearly growing input of the problem. As pointed out in section 4.2 the amount of computations necessary, in case of a naive algorithm, grows exponentially if the path length increases by one hop, therefore we consider our problem as complex. Further if a complex problem has the property of overlapping sub problems [26] a recursive algorithm can solve the same sub problem repetitively and memorize the results for later lookup until the complex problem is solved by an eventual combination of the results of the sub problems. This procedure takes considerably less computational time than a naive method, which does not take advantage of the sub problem overlap.

The standard procedure named the *Viterbi algorithm* is proposed in [87]. The following paragraph briefly illustrates the principle of the Viterbi algorithm applied on a first order Markov chain for obtaining a minimal travelling path.

Let x be a sequence of length n and c_i from the label alphabet S with $i = 1, \ldots, n$, we want to find a setup c^* that minimizes the total costs:

$$Score(x, c) = \sum_{i=1}^{n-1} (e_i^{c_i} + t_{i,i+1}^{c_i, c_{i+1}}) + e_n^{c_n} \tag{4.4}$$

where $e_i^{c_i}$ is the emission scoring for hop x_i having a state of c_i, and $t_{i,i+1}^{c_i,c_{i+1}}$ is the scoring of transitioning from state c_i to c_{i+1}.

Following the idea of dynamic programming, for all $1 \leq i \leq n$ we can compute the evaluation $v_{c_i}(i)$ for the subsequence getting state c_i at time i by iteratively computing Equation 4.5.

$$v_{c_i}(i) = e_i^{c_i} + \min_{c_{i-1} \in S} \{v_{c_{i-1}}(i-1) + t_{i-1,i}^{c_{i-1} c_i}\} \tag{4.5}$$

where c_{i-1} is a possible state of the sub-sequence at time $i-1$. We can reconstruct the path c^* itself by keeping back pointers during the recursive stage and tracing them. Viterbi algorithm for a Markov chain of first order takes a time complexity of $O(n \cdot |S|^2)$ with a space consumption of $O(n \cdot |S|)$.

However, if we increase the order of the Markov property to m, we have to memorize up to m previous states which requires a cost of $O(n \cdot |S|^m)$ on storage demands. Furthermore, an m-th order Markov chain over the alphabet S is shown to be equivalent to a first order Markov chain over the alphabet S^m of m-tuples [70]. This means, the time complexity of Viterbi algorithm for a such model takes $O(n \cdot |S|^{2m})$. For our application we have an m order Markov chain. In section 4.4, we give a detailed example of the application of Viterbi on a m order Markov chain, where in our case $m = 2$, which finds c^*.

We reserve the implementation of Viterbi for future work and design a simple approximative greedy algorithm in order to evaluate the capability of RDV beforehand. The greedy principle in general is an optimization idea by making the locally optimal choice at each stage [61]. In most cases, algorithms based on the greedy principle would not guarantee a globally optimal solution, but they provide an approximation in a reasonable time.

The basic idea of the *Greedy* channel selection algorithm is to separately find an optimal channel combination for a sub set of the paths hops and then determine the remaining channel selections based on the first subset. We call this subset a window through which we examine the path. In general window limits the view on the whole object, we use this to limit complexity to heuristically find a solution. We fix a Window Size (WS) and focus it on one end of the path and compute all possible channel combinations within this window using the naive algorithm and equation 4.1. After the best channel sequence of length WS is determined, it is fixed for the first WS hops. For the remaining $n - WS$ hops we slide the window by one hop towards to other end of the path and greedily chose the best channel out of $|S|$ channels. The algorithm is described in further detail in section 4.5. The window concept is inspired by the first step of a m-order Viterbi algorithm, where for the first $m + 1$ segment of the path all possible combinations are computed. The *Greedy* algorithm treats the problem as first order after determining the minimum for the first WS states. Therefore, we can expect a complexity of $O(|S|^W S + (n - WS) \cdot |S|)$ for computation time and a space consumption as same as the first order case by $O(n \cdot |S|)$.

Having done this, we can theoretically expect an approximation error Err_{greedy} of

$$\text{Err}_{greedy} < \sum_{i=WS+1}^{n} \text{Err}_{max}(h_i), \tag{4.6}$$

where $\text{Err}_{max}(h_i)$ denotes the maximal error by the i-th hop in the routing path.

Moreover, if we use the *Diverse algorithm*, which assigns the labels from the alphabet $|S|$ in a consecutive order e.g. A, B, C, A... We can estimate the error $\text{Err}_{diverse}$ for this approximation as follows:

$$\text{Err}_{diverse} \leq \sum_{i=1}^{n} \text{Err}_{max}(h_i) \tag{4.7}$$

It seems, that the errors in Equation (4.6) as well as in Equation (4.7) do not differ that much from each other.

Nevertheless, if we make a more detailed analysis, we will spot a clear difference. In section 5.1 we evaluate the solution quality of our approximative algorithms *Greedy* and *Diverse* numerically and compare it to the optimum which the Viterbi algorithm will deliver in future work.

Notice that for a uniform distribution of the emission scores e (which represent the estimated channel metrics) we can expect much less error using Greedy algorithm, since we calculate the minimum at each iteration compared to Diverse algorithm, in which we are forced to choose a certain channel combination.

It is worth to mention, that the choice of the window size for our Greedy algorithm, which depends on the interference range in the network, represents the Markov property in the given sequential graph model. Under our assumption for an interference range of 3, it is suggested to choose a window size of 3 for the study. In the following section we will specify the second order Markov chain Viterbi algorithm and provide an example for its functionality.

4.4 The Viterbi Algorithm

For a better understanding of how the Viterbi algorithm can solve the channel selection problem described in section 4.2 we first give an analogy from a speech recognition application and then an example based on our four hop path shown in figure 3.2.

The Viterbi algorithm is used for a variety of problems where an input sequence of observed tokens needs a label from a certain alphabet, the output sequence of most likely hidden states, if the label depend on previous labels (Markov Chain property). The applications vary from encoding of convolutional codes used for error correction in Groupe Special Mobile (GSM), Code Division Multiple Access (CDMA), satellite communication, hard drives or 802.11 to bioinformatics and speech recognition.

Lets assume there is an acoustic signal as depicted in figure 4.7 which when played back sounds like a word e.g. 'hello' and the goal of the exercise is to decode the signal into a sequence of letters, speech-to-text. For a machine the the acoustic signal is a series of observed tones of certain frequencies, therefore the real sequence of letters is hidden. The Viterbi algorithm can be used to find the most likely sequence of letters Y from a finite alphabet S hidden behind the observed acoustic signal X of length n by using probabilities of occurrence of letters. The algorithm needs three types of probabilities.

Initial probabilities π which is an array of size |S|, they describe how likely it is that e.g. the letter 'e' occurs or the letter 'x'. Where $Pr(letter\, e) > Pr(letter\, x)$ in the English language. Further it needs emission probabilities E, which is a matrix of size $|S| \cdot n$, they describe how likely it is that e.g. a tone is the letter e. Finally it needs transition probabilities T, which is a matrix of size $|S| \cdot |S|$, they describe how likely it is that e.g. a letter follows a previous letter, where $Pr(e\, follows\, l) > Pr(s\, follows\, x)$ in the English language. The probabilities are obtained beforehand through means of algorithmic training. The input/output relation for this analogy is summed up as follows:

$$Viterbi(X, S, \pi, E, T) : Y$$

In our case the observed sequence of tokens X are the four hops (length n) between the five nodes in the example in figure 3.2. The Viterbi algorithm can be used to

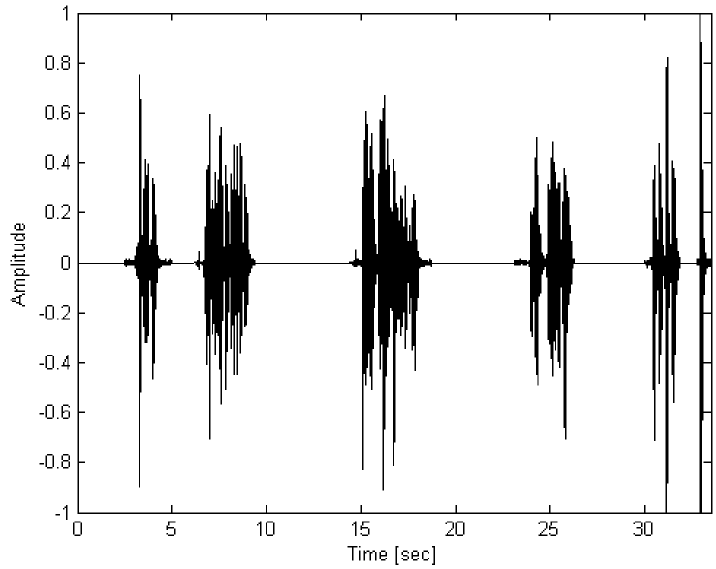


Figure 4.7: Acoustic Signal

find sequence of channels C (instead of Y in speech-to-text analogy) from the finite alphabet S with the minimal path metric. Where S is the set of channels available at every hop, in our case channel A, B, C. The algorithm in our case does not work with probabilities, but with values of Expected Transmission Time (ETT) or ALM to be precise. To find the channel sequence combination with minimal ETT and minimal IaFI it needs two types of scores (instead of probabilites). Emission scores E, which is a matrix of size $|S| \cdot n$, it contains the channel metrics values for each link along the path derived from the route record. For instance the channel metric for hop x_3 between node I_2 and I_3 on channel C is 180. Transition scores T, which is a matrix of size $|S| \cdot |S| \cdot |S|$, it contains the IaFI penalties. The input/output relation is summed up as follows:

$$Viterbi(X, S, E, T) : Y$$

Notice, contrary to the speech to text case, we do not have any input parameters in analogy to initial probabilities. As mentioned previously we assume R_I of 3 hops, which means that e.g. hop x_1^A interferes with hop $x2^A$ and hop x_3^A but not with hop x_4^A. For the Viterbi algorithm this means that the Markov property is of second order. Further we consider the emission features $e_i^{c_i}$ as the estimated channel metric for the communication on $hop_{x_i x_{i+1}}$, while the penalties of using the same channel within interference range are described by the transition features, which are defined for $i = 2, \ldots, n-2$ by equation 4.8:

$$t^{c_{i-1}c_ic_{i+1}}_{i-1,i,i+1} = \begin{cases} 2 \cdot e^{c_{i-1}}_{i-1} + e_i{}^{c_i} + e^{c_{i+1}}_{i+1} & for\, c_{i-1} = c_i = c_{i+1} \\ e^{c_{i-1}}_{i-1} + e^{c_l}_l & for\, c_{i-1} = c_l \neq c_m \\ & l, m \in \{i, i+1\} \\ 0 & \text{otherwise} \end{cases} \tag{4.8}$$

We iteratively compute equation 4.9.

$$v_{c_i,c_{i-1}}(i) = e^{c_i}_i + \min_{c_{i-1},c_{i-2} \in S^2} \{v_{c_{i-2},c_{i-1}}(i-1) + t^{c_{i-1},c_i,c_{i+1}}_{i-1,i,i+1}\} \tag{4.9}$$

To find the Viterbi path (optimal channel sequence) at the end of the computation we must save back pointers, that remember which channel was used in equation 4.9 as minimal score at every step.

In what follows is an example calculation of the Viterbi algorithm on our four hop example path. In table 4.2 the first column represents a 'line' number 1 to 27. The amount of lines is a result from the number of possible channel combinations for the first three hops if three channels are available, $3^3 = 27$. The column 'Prefix' denotes the 27 different channel combinations from the alphabet S. The first combination in line 1 is AAA, which means that channel A is used on hop x_1, x_2, and x_3. The last combination in line 27 means that channel C is used on all three hops.

Column '$x_1x_2x_3$' represents the Viterbi evaluation for each of the 27 possible channel combinations for the initial three hops. According to equation 4.8 the focus for the calculation lies on hop x_2. The algorithm computes the evaluation as a sum of the estimated channel metrics, which we call emission features to conform with terminology commonly used with the Viterbi algorithm, plus a penalty if a channel is reused as defined in that equation. For instance the evaluation (or score) for the Prefix AAA is the sum of the emission features for the first hop x_1 using channel A e^A_1=100 plus the second hop x_2 e^A_2=100 plus the third hop x_3 e^A_3=120 plus the penalty for that Prefix t^{AAA}_{123}=$2 \cdot 100 + 100 + 120 = 420$. This step is performed for all 27 Prefixes of the first three hops. For the consideration of the channel for the forth x_4 only hop x_3 and hop x_2 matter, because R_I is 3 hops, hop x_1 does not matter. Because only the last two channel labels AA of the Prefix AAA are relevant, we define the Suffix as the last two labels of the Prefix and display it in column 'Suffix'. The Suffix consists of two hops with three possible channels, therefore $2^3 = 9$ unique combinations such as AA, AB, ... , CC.

The next step of the Viterbi algorithm is the identification of the minimal cost for a given Suffix for the Markov chain x_1, x_2, x_3. For example for the Suffix AA, which means that hop x_2 and hop x_3 both use channel A, three unique Suffixes exist, AAA=740, BAA=545 and CAA=640. Thus the minimal cost Viterbi score is

Table 4.2: Example Viterbi

Line	Prefix		Suffix	$X_1\ X_2\ X_3$	min. to X_3 with same suffix	$X_2\ X_3\ X_4$	min. to X_4 with same suffix	$X_3\ X_4\ X_5$
1		A	AA	$e_1^A+e_2^A+e_3^A+t_{123}^{AAA}=740$	$V_{AA}(3)$=545	$V_{AA}(3)+e_4^A+t_{234}^{AAA}$=1065	$V_{AA}(4)$=675	
2	AA	B	AB	$e_1^A+e_2^A+e_3^B+t_{123}^{AAB}=510$		$V_{AA}(3)+e_4^B+t_{234}^{AAB}$=870		$V_{AA}(4)+e_5^\tau+t_{345}^{AA\tau}$=895
3		C	AC	$e_1^A+e_2^A+e_3^C+t_{123}^{AAC}=580$	from Line 10	$V_{AA}(3)+e_4^C+t_{234}^{AAC}$=965	from Line 19	
4		A	BA	570	$V_{AB}(3)$=480	780	$V_{AB}(4)$=460	
5	AB	B	BB	580		800		$V_{AB}(4)+e_5^\tau+t_{345}^{AB\tau}$=460
6		C	BC	410	from Line 20	680	from Line 20	
7		A	CA	570	$V_{AC}(3)$=385	685	$V_{AC}(4)$=650	
8	AC	B	CB	340		490		$V_{AC}(4)+e_5^\tau+t_{345}^{AC\tau}$=645
9		C	CC	720	from Line 12	965	from Line 12	
10		A	AA	545	$V_{BA}(3)$=450	770	$V_{BA}(4)$=440	
11	BA	B	AB	530		790		$V_{BA}(4)+e_5^\tau+t_{345}^{BA\tau}$=440
12		C	AC	385	from Line 22	650	from Line 22	
13		A	BA	590	$V_{BB}(3)$=580	920	$V_{BB}(4)$=440	
14	BB	B	BB	795		1160		$V_{BB}(4)+e_5^\tau+t_{345}^{BB\tau}$=875
15		C	BC	650	from Line 5	1020	from Line 23	
16		A	CA	355	$V_{BC}(3)$=410	510	$V_{BC}(4)$=440	
17	BC	B	CB	560		750		$V_{BC}(4)+e_5^\tau+t_{345}^{BC\tau}$=675
18		C	CC	725	from Line 6	990	from Line 6	
19		A	AA	640	$V_{CA}(3)$=355	675	$V_{CA}(4)$=440	
20	CA	B	AB	480		460		$V_{CA}(4)+e_5^\tau+t_{345}^{CA\tau}$=510
21		C	AC	860	from Line 16	885	from Line 16	
22		A	BA	450	$V_{CB}(3)$=340	440	$V_{CB}(4)$=490	
23	CB	B	BB	680		660		$V_{CB}(4)+e_5^\tau+t_{345}^{CB\tau}$=490
24		C	BC	890	from Line 8	870	from Line 8	
25		A	CA	780	$V_{CC}(3)$=720	1130	$V_{CC}(4)$=965	
26	CC	B	CB	770		1135		$V_{CC}(4)+e_5^\tau+t_{345}^{CC\tau}$=1340
27		C	CC	1220	from Line 9	1560	from Line 9	

$Vc_{i+1}, c_{i+2}(i+2) = V_{AA}(3) = 545$ found in line 10.

What follows is the evaluation of hop x_4, which has the choice of using one of three available channels with each of the 9 Suffixes from the previous three hops, so that again 27 different scores are computed, this is represented in column $x_2x_3x_4$. For example the score for the sequence $AAAA$ is equal to score $V_{AA}(3) = 545$ plus the channel metric for hop x_4^A=100 (also called emission feature e_4^A) plus the penalty if channel A is reused $t_{234}^{AAA} = 2 \cdot 100 + 120 + 100 = 420$ (sum 1065). Further again the minimal cost for each of the 9 unique Suffixes is determined and listed in column 'min. to x_4 with same suffix', e.g. $V_{AA}(4) = 675$, found in line 19.

Notice that the last evaluation step that we demonstrated for hop x_4 will be repeated for every next hop for longer chains, by re utilization of the previous step with the minimal score for the two previous channels (Suffix). This repetitive solution of the same sub problem gives the Viterbi algorithm its recursive property. We also want to point out that because only the last two channel combinations are relevant for the consideration of the next hop (second order Markov property) the algorithm drastically limits the amount of computations compared to a naive algorithm described in section 4.2.

Further if hop x_4 is the last hop in the chain, we need to complete the evaluation with another step to finally consider all penalties. To achieve that we use an artificial (non existing) hop x_5^τ with an imaginary channel τ which is different from the channels A, B, C from alphabet S. For example the final evaluation score for hop x_4 is $V_{AA}(4) = 675$ plus e_5^τ=0 plus $t_{345}^{AA\tau} = 220$, which results in 895.

With the evaluation complete, the algorithm determines the minimal score found in column $X_3X_4X_5$ of 440 for the second order Markov chain x_1, x_2, x_3, x_4 with the state set S and the estimated channel metrics (emission features) from figure 3.2. To find the best channel sequence the algorithm applies backtracking, which means that back pointers for the minimal score for each Suffix at each hop are memorized. In our case the minimal Viterbi score $v^* = v_{BA}(4) = 440$ indicates that for the last two hops x_3, x_4 the channels BA should be used. We can then recursively find that $v_{BA}(4) = v_{CB}(3) + e_4^A + t_{2,3,4}^{CBA}$. This indicates that for the hop x_2 channel C should be used. Further $v_{CB}(3)$ is the minimal score found in line 8, which indicates that for hop x_1 channel A should be used. Through this iteration we trace the path and obtain a Viterbi solution with the optimal (in terms of channel metrics) and IaFI minimal channel sequence $c^* = ACBA$.

4.5 Greedy Channel Selection Algorithm

To optimize channel selection using equation 4.1, it is necessary to compute the cost for any possible channel combination in a path and select the combination with minimum cost. However the time complexity for that is exponential. As described in section 4.3, it is possible to reduce the complexity by taking advantage of the fact that the interference range is limited and applying the Viterbi algorithm to find the best combination in polynomial time. To minimize the time and storage complexity further, we reduce this problem to finding a sub-optimal channel combination to gain a practical advantage over necessary computation overhead. As explained in section 4.3, a window size of at least $WS = 3$ would be enough for the Greedy algorithm, which works as follows.

For the first step we start from the beginning of the path and select the channels for first WS hops (chain x_1, x_2, x_3) by minimizing the cost over all possible channel combinations in the window. To achieve that we generate a matrix with all possible channel combinations for the first WS hops. The size of the matrix is $WS^{|S|}$, for instance if window size is 3 and the number of channel is 3, the matrix has the size of 27 unique channel combinations. Algorithm 1 presents pseudo code which only describes some important functions, the detailed programming code however is out of scope of this work.

In general the function 'SelectOptimalChannelSequence' takes as input the window size, the number of common channels between each node and a route record vector R for the first WS hops of the path. It then generates a matrix of unique channel combinations *c_matrix* and a matrix of IaFI penalty factors *p_matrix*. The penalty factor matrix contains a factor for each link of every unique channel sequence computed according to the condition in equation 4.1. Further *m_matrix* which contains the channel metrics for each channel is created using *c_matrix* and the route record vector R. Also *e_matrix* is created as a product of *c_matrix* and *p_matrix*, which is the evaluation matrix, containing the channel metric values after consideration of IaFI. Finally by computation of the sum of every row of *e_matrix*, the evaluation result m_c for each unique channel sequence is obtained and stored. The optimal channel sequence is then retrieved and used to set the channel selection flag and penalty factor in the route record for the first WS hops. The route record R is then marked as selected R_s and returned by the algorithm.

This initial step similar to the naive approach, but limited to the first WS hops of the path. It also resembles the initial step of the Viterbi algorithm with respect to time complexity. Table 4.3 shows the calculations of the first step for the example in figure 3.2.

Algorithm 1 Optimal Channel Sequence Computation

Input:
R: route record vector of first WS hops;
S: Set of common channels;
WS: Windows size;
Output: R_s route record with set channel selection flags and penalty factors

Description:

```
function GenerateChannelAndPenaltyMatrices(S,WS)
    c_matrix = 0                                   ▷ matrix of unique channel combinations
    p_matrix = 0                                   ▷ matrix of penalty factors
    for |S|^WS do
        c_matrix ← GenerateUniqueChannelSequence(...)
        p_matrix ← ComputePenaltyFactor(...)
    end for
    return c_matrix, p_matrix
end function
function SelectOptimalChannelSequence(S,WS,R)
    o_vector = 0                                   ▷ vector with optimal channel sequence
    c_matrix, p_matrix ← GenerateChannelAndPenaltyMatrices(S,WS)
    m_matrix ← CreateMetricMatrix(c_matrix, R)     ▷ symmetric to c_matrix
    e_matrix = 0                                   ▷ evaluation matrix
    for i ≤ |S|^WS do
        for j ≤ WS do
            e_matrix[i][j] =m_matrix[i][j]·p_matrix[i][j]
        end for
    end for
    for every row in e_matrix do
        compute sum of columns
        append sum to end of row
    end for
    sort e_matrix in increasing order
    o_vector ← e_matrix[1]                         ▷ set optimal channel sequence vector
    R ← SetOptimalChannelSequence(o_vector)        ▷ set selection flags in R
    return R_s
end function
```

Table 4.3: Example Greedy Step 1

Combination	m_c	**Combination**	m_c
AAA	1280	BBC	885
AAB	710	BCA	355
AAC	780	BCB	775
ABA	790	BCC	1035
ABB	820	CAA	880
ABC	410	CAB	410
ACA	790	CAC	1240
ACB	340	CBA	450
ACC	1030	CBB	920
BAA	765	CBC	1270
BAB	745	CCA	1210
BAC	385	CCB	1100
BBA	825	CCC	2040
BBB	1380	-	-

The optimal channel sequence is determined as ACB and a metric of $m_{c_{123}^{ACB}} = 340$. However in the next step we fix the chosen channels for each hop and slide the window by one hop to the chain x_2, x_3, x_4 and compute the cost for all $|S|$ possible combinations and select the combination with the minimal metric m_c as shown in table 4.4.

Table 4.4: Example Greedy Step 2

Combination	m_c
CBA	340
CBB	775
CBC	1085

The channel sequence CBA with the minimal cost is, $m_{c_{234}^{CBA}} = 340$. This determines the use of channel A on $link_{x_3x_4}$. The final channel sequence for the chain x_1, x_2, x_3, x_4 is $ACBA$ an the corresponding path metric $m_{c_{1234}^{ACBA}=440}$. Notice the result is the same as the result of the Viterbi algorithm thus the optimal combination on this 4 hop long example path. However the Greedy algorithm is expected to deliver suboptimal solutions for longer paths, because its approximation error increases with the path length as described in section 4.3. Contrary to the Viterbi algorithm this second step only considers $|S|$ choices instead of $|S|^2$ choices and does not require memorization of sub problem results for later lookup because the selected channels are fixed using the channel selection flag of the route record. In the same way, by repeating the second step, a channel can be selected on longer paths for all hops by sliding the window along the path until the end.

A larger window size has the advantage of a more accurate optimization due to covering more number of links at the beginning. According to equation 4.6, this would reduce the expected approximation error. For our experiments we use a window size of 4, to balance the time and storage complexity and expected approximation error.

Compared to the Greedy algorithm, the Diverse channel selection algorithm simply selects the channel sequence $ABCA$ irrespective of the channel metrics on the available links. To reflect its ignorance of ETT, thus prevailing transmission conditions on the channels we arbitrarily decide to chose the first channel A on the first hop, the second channel B on the second, the third channel C on the first, the first A channel on the fourth hop and so on. Its channel sequence metric score is $m_{c^{ABCA}_{1234}=510}$, which has minimal IaFI, but not optimal with regard to channel characteristics. However it has negligible computation demand and therefore an interesting option for nodes with low processing and memory resources.

4.6 Considerations for Practical Implementation

RDV can be implemented on multi-radio platforms which support the IEEE 802.11s standard. Figure 4.8 displays a rudimentary class diagram of the HWMP protocol as implemented in NS-3. A more detailed class diagram can be found in appendix A.1. Every node executes an instance of the $HwmpProtocol$ class, which implements the path formation mechanism and route lookup mechanism. The path formation mechanism uses the procedures $ReiceivePreq()$ and $ReceivePrep()$ to process PREQ and PREP packets.

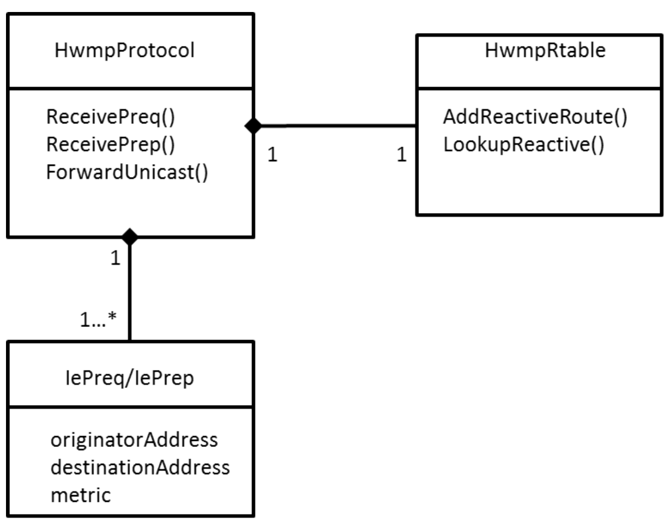


Figure 4.8: HWMP Class Diagram

Figure 4.9 shows a simplified state machine of the HWMP protocol. Whenever a PREQ packet is received, it is checked if it carries fresh information with respect to its sequence number or if its metric field contains a better value than an already known path. If accepted a routing table entry is created or updated and the frame is forwarded (in case of an intermediary node). A routing table object has exactly one routing table object termed *HwmpRtable* in figure 4.8. Further it can be related to multiple $IePreq$ and $IePrep$ objects which constitute PREQ and PREP frames.

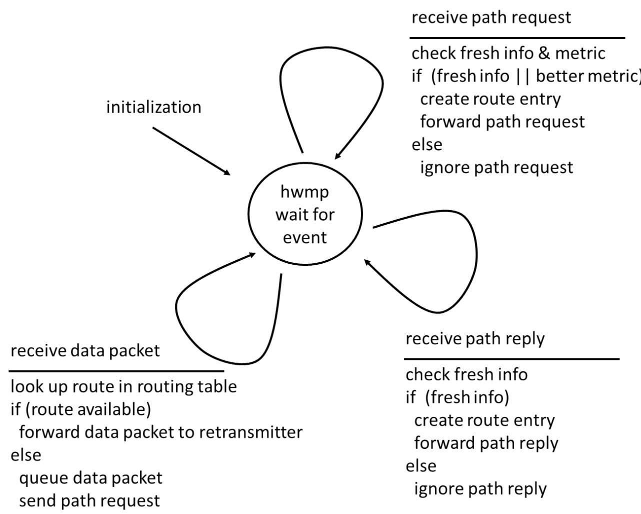


Figure 4.9: HWMP State Machine

The necessary modifications in protocol behaviour as described in section 4.1 for the implementation of RDV must be done in the *HwmpProtocol* class as figure 4.10 demonstrates. We term the class *RdvProtocol* and modify the *ReceivePreq*() and *ReceivePrep*() procedures. The asterisk $*$ marks modified or new procedures and objects. We introduce two classes *RouteRecordUnit* and $InterfaceUnit$. As shown in figure 4.4 a PREQ and PREP packet has a vector of one or more route record entries. Correspondingly an $IePreq/IePrep$ object is related to one or more *RouteRecordUnit* objects. Similarly a route record has multiple interface entries as shown in figure 4.5, so that a *RouteRecordUnit* object is related to multiple $InterfaceUnit$ objects. To realize this structure also serialization and deserialization procedures must be adapted to allow route record to be successfully decomposed at the sender, transmitted and recomposed at the receiver.
The *ReceivePreq*() and *ReceivePrep*() procedures must be adopted as described in figure 4.11, so that a route record unit is appended to the PREQ packet at every node and that not routing table entry is made at intermediary nodes during the broadcast phase.

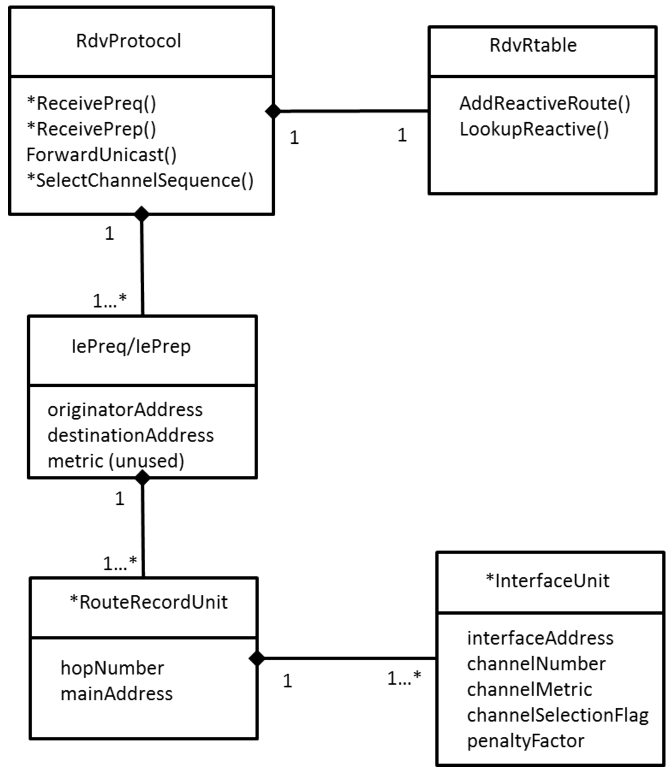


Figure 4.10: RDV Class Diagram

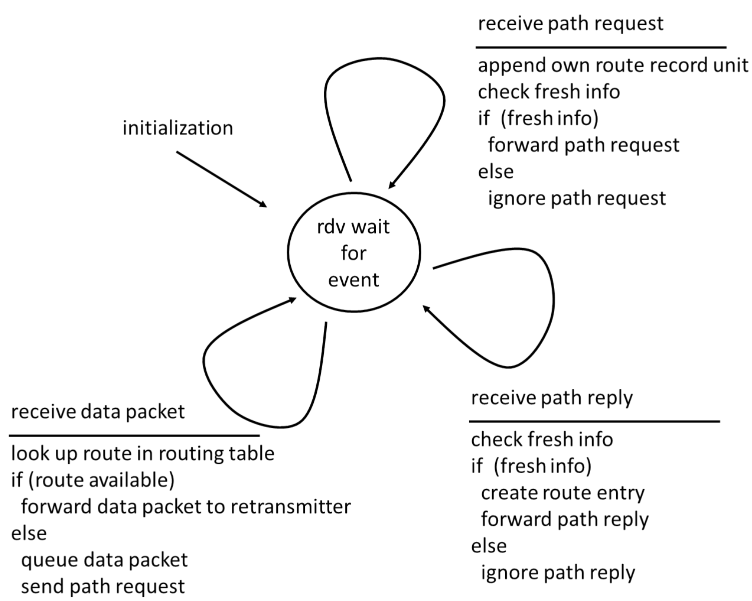


Figure 4.11: RDV State Machine

We also introduce the procedure $SelectChannelSequence()$, which optimizes the channel sequence of the discovered path according to our proposed algorithms Diverse, Greedy, Viterbi (in future work) or any other optimization algorithm. Notice that the integrity of the routing table class is not compromised. We assume that a real IEEE 802.11s such as [44] implementation will be comparable the NS-3 implementation if it is realized according to standard specification and should be feasible with moderate amount of software development.

4.7 Chapter Summary

This chapter presented the design for our proposed routing protocol RDV in comparison with AODV and HWMP respectively. Further it described the channel selection problem which results from the protocol design. The channel selection problem was found to be complex because its computational demand increases exponentially with the linear increase of its input, the path length. We investigated the problem theoretically and found that the fact that interference range is limited can be exploited to reduce the complexity of the problem by applying a recursive algorithm to solve the problem with reasonable time and memory demand. Further we described a Greedy algorithm that solves the problem with even less demand sub optimally and provided a consideration for a practical implementation of our protocol on real hardware nodes. In the following chapter we will gather empirical evidence with regard to our proposed routing concept and algorithms.

CHAPTER 5

Performance Evaluation

Contents

5.1 Numerical Evaluation

As mentioned before, the Greedy channel selection algorithm results in selecting a channel combination with a sub-optimal cost. In fact, as section 4.3 describes in details, there is a trade-off between the cost optimization capability of Greedy algorithm and its required computational power. In this section we evaluate the ability of the Greedy algorithm with regard to cost optimization. We compare its output to the optimum channel combination which the Viterbi algorithm can find, using a numerical implementation in Matlab. Given a multi-channel path, similar to figure 3.2 but longer, we generate all possible channel combinations and calculate their costs using equation 4.1 with the naive computation approach. We find the optimal channel combination by searching the minimum cost combination among them. As discussed in section 4.2 the naive algorithm has a exponential time and memory complexity of $O(|S|^n)$. This means it would not find the optimal solution in reasonable time on a typical embedded device of todays standards such as [63] with three interfaces and channels for larger networks.

We also do the same evaluation for the Diverse channel selection algorithm. As Diverse algorithm does not consider any channel metric values to optimize the channel sequence, its result is not expected to be necessarily close to the optimum channel combination in which the cost is optimized considering both channel diversity and metric values.

In this numerical study, a route of length $n = 10$ hops with randomly generated metric values for the 3 interfaces at every hop is considered. Then the path optimized by Greedy algorithm is found, $Path_{greedy}$. Also, all possible channel combinations will be generated and stored along with their total cost values in a large matrix, M, which thus will have 3^n rows. We sort the rows of this matrix regarding the cost values in an increasing order. To evaluate the result of Greedy algorithm against the optimum path (which would be the first row of the matrix M), we look up $Path_{greedy}$ in matrix M and use the row number of the found match, called $rank$, as a measure of the optimization performance of the algorithm.

We perform the above evaluation with two different metric distributions, corresponding to two different situations regarding the pattern of links metrics in the network. The first case is if the network is not very busy and thus the high metric values appear rarely and only on the link which are used by multiple flows. We model this case by generating the metric values using Poisson random generator, which would assign the metric values around its λ parameter to the majority of links and the large metric values far from λ would appear with a very small chance on the links.

The second kind of situation is if the network is very busy (large number of flows and/or high bit rates). In this case, more links would have large metric values (the links used by high bit rate flows or the ones used by multiple flows). In our simulations, this is represented by uniformly distributed metric valued generated by a uniform random generator.

For the Poisson distribution we use $\lambda = 150$ and the uniform distribution generates random metric values in the range [150 800]. The simulation result is the average over 300 runs.

Figure 5.1 presents the average $rank$, as defined above, for Greedy and Diverse algorithms. This figure shows that with a uniform distribution for metric values the difference between performance of Greedy and Diverse algorithms would not be as large as for Poisson distribution. This is because, in the first scenario metric values are uniformly distributed and therefore the advantage brought by the ability of Greedy algorithm in taking metric values into account would not be as significant as the first scenario. In the second scenario, most of the links take metric values around the mean of the distribution (Poisson behaviour) and only a minority are having high cost values and thus the main role in optimizing the total cost is played by channel diversity which is provided by Diverse algorithm too.

As this numerical study suggests, we can expect a larger difference between the performance of Greedy and Diverse if the network traffic results in metric values well distributed in a range and not all around the mean. However with both distributions the result of the Greedy algorithm is very close to the optimum combination (rank 1), hence to the performance of Viterbi with less time and storage complexity. Further the distribution of channel metric values, therefore traffic patterns play a large role.

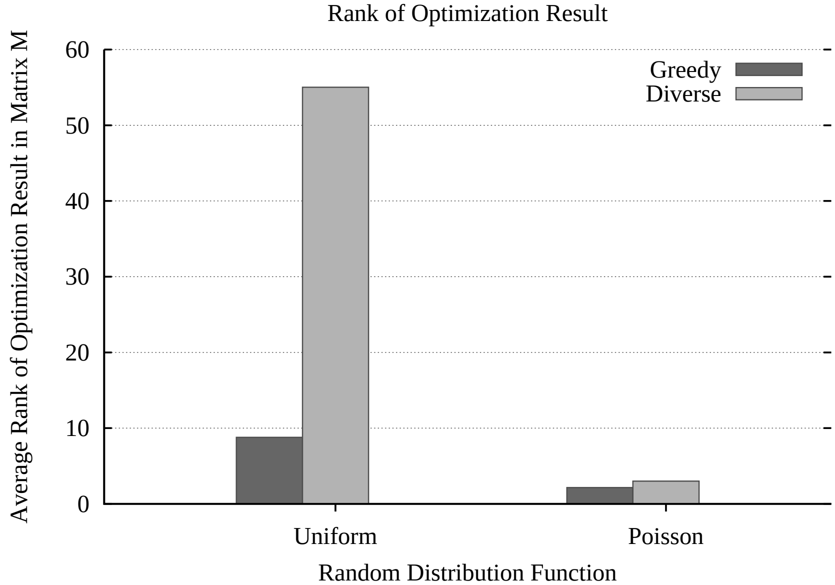


Figure 5.1: Rank of optimization result.

For low loads (Poisson) the Diverse algorithm offers good optimization at virtually no cost. For higher loads (Uniform), Greedy algorithm offers an optimization advantage at good efficiency. To justify the implementation of the Viterbi algorithm, the channel metric distribution should be closer to the uniform distribution.

5.2 Simulative Evaluation

This section presents our empirical findings by simulation using the discrete event NS-3 [28]. Our work is based on an implementation of the 802.11s draft standard [88], which supports multi-radio, multi-channel nodes. A thorough introduction of 802.11s D3.0 implementation can be found in [4]. We consider a multi-hop topology placed as a chain or grid with a single or multiple simultaneous connections. Our experiments compare our proposed protocol RDV with its according metric with HWMP and the following metrics ALM, WCETT and CIETT.

5.2.1 Communication and Interference Range

In this section we determine the distance with which two nodes with a single hop can communicate with each other and at what bit rate using a single interface and channel. To achieve that, we have set up two nodes on which a UDP constant bit rate sink and source are installed as applications. One node sends UDP packets of size 1024 Kilo Byte (Kb) at a constant rate to the other using IEEE 802.11a for the physical layer, which means the spectrum between 5.2Ghz and 5.8Ghz is used to modulate binary data. At 1 metre distance between the nodes the maximal received bit rate is about 23.5 Mbps, from that we assume that the modulation scheme used by the physical layer at this distance is Orthogonal Frequency-Division Multiplexing (OFDM) with 16 Quadrature Amplitude Modulation (QAM) with 48Mbps gross bit rate. We set the transmission bit rate of the UDP sink at 24Mbps and subsequently increase the distance by one metre until no communication is possible. Figure 5.2 shows the average received bit rate at the sink node at any given distance. The transmission duration is 30 seconds, we repeat the transmission at every distance 30 times to account for variation in the signal propagation model of NS-3. However the variation is negligible and no error bars appear on the plot.

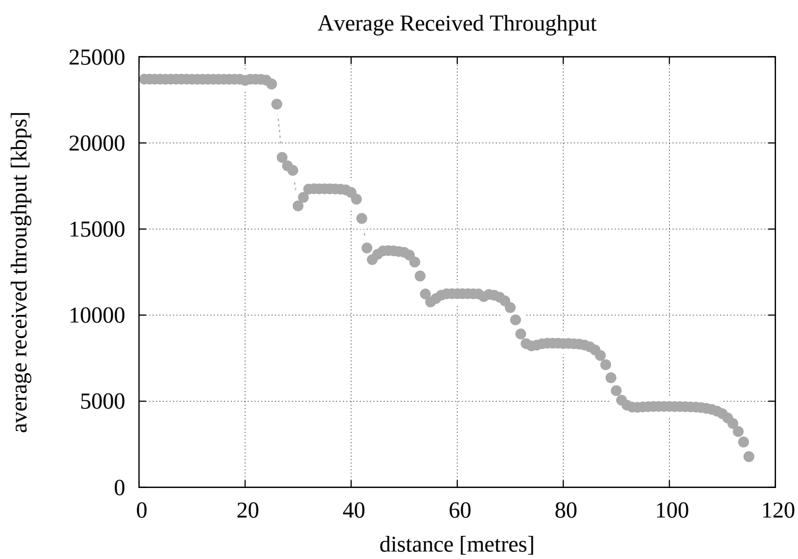


Figure 5.2: Average received throughput at varying distance.

One can see that at the distance increases between the nodes, the received bit rate decreases with 6 steps, but stays constant within a certain range. The reason for this is as mentioned in section 2.1.1 the decrease of the received signal power with distance according to the inverse square law. For instance in the range of 1-25 metres according to [80] 16 QAM is used with a net bit rate of about 24 Mbps, however as the signal deteriorates further in the range of 26-31 metres, the net bit rate falls sharply from around 24Mbps to about 16 Mbps due to a high amount of bit errors. The reason is that 16 QAM in our environment at this distance is to sensitive for low signal power in relation to noise due to its high density of coded bits. As a result the physical layer changes the modulation scheme from 16 QAM to Quadrature Phase-Shift Keying (QPSK) (which has a less dense bit coding scheme) with a net bit rate of about 18 Mbps at the distance of 32 metres. Table 5.1 shows inter node distance and assumed modulation scheme and bit rates. The modulation schemes are assumed, because the observed received bit rates indicate them, we have no means or record from our environment which modulation scheme was used when.

Table 5.1: Internode distance, bit rates and modulation schemes.

Distance	**Net Bit Rate**	**Gross Bit Rate**	**Modulation**
1-31	24 Mbps	48Mbps	16 QAM
32-45	18 Mbps	24Mbps	QPSK
46-56	14 Mbps	24Mbps	QPSK
57-75	12 Mbps	24Mbps	BPSK
76-88	9 Mbps	12Mbps	BPSK
89-120	6 Mbps	12Mbps	BPSK

Notice that for certain gross bit rates different net bit rates exist. This is due to different amounts of error correction redundancy per frame, for more information on this refer to the specification [80]. Also notice the range of 46-56 metres in our observations where we record received bit rates of about 14 Mbps does not exist in the specification, it can be assumed that in this range heterogeneous bit rates are used on OFDM sub carriers. At the distance of 121 metres no communication was recorded, from which we assume that this is the maximal Communication Range (R_C).

From these observations we find that the maximal Communication Range (R_C) with a useful bandwidth is in the area of 89-112 metres. Therefore we chose an inter node (hop) distance of 100 metres for our further experiments. At this distance the signal strength from more distant hops is minimal, which helps to reduce interference. We chose a UDP source application transmission bit rate of 4550 Kilo Bit per Second (Kbps) as the optimal test bit rate at which the following transmission metrics are observed on a single hop under perfect conditions, meaning no surrounding network activity, as shwon in table 5.2. Notice that the received bit rate (Rx) is higher than the transmitted bit rate (Tx). We set the transmission bit rate at the application layer, but record the received bit rate at the transport layer, therefore the received bit rate also includes IP overhead, which explains the discrepancy.

Table 5.2: Ideal Link Performance at 100 Metres

Tx Bitrate	**Rx Bitrate**	**Packet Delay**	**Packet Delivery Fraction**
4550 Kbps	4674 Kbps	4 ms	1.0

As next step we determine the Interference Range (R_I) of our environment in terms of number of hops if hop length is 100 metres. For this a simple experiment is set up with four nodes as depicted in figure 5.3. The nodes have one single interface and channel. Node n_1 sends data at 4550 Kbps to node n_2, both nodes are one hop, 100 metres apart. Node n_3 and n_4 do the same at the same time and are also 100 metres apart. To find the interference range we vary the distance d between n_2 and n_3 and record received throughput, packet delay and the fraction of successfully delivered packets of all packets sent. Every transmission takes place for 30 seconds and is repeated 30 times, the reported performance metrics are average values.

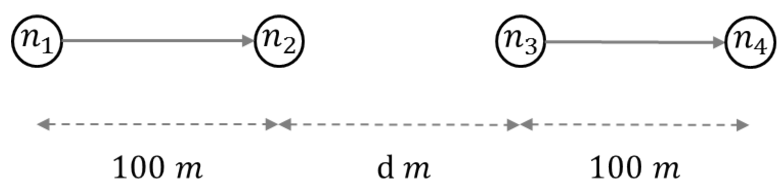


Figure 5.3: Interference Range Experiment

First $d = 100m$ is set up to observe the impact of n_3s transmission on n_2s reception if n_3 is one hop away. Figure 5.4 shows the received throughput at 100 metres at about 2.9 Mbps of the ideal 4.6 Mbps, which indicates that n_3s transmission does interfere with n_2s reception. Then d is increased to 200 metres, or two hops which records 3.7 Mbps of 4.6 Mbps. Further d is increased to 300 metres, or three hops and n_2 received the ideal bit rate of about 4.6 Mbps. To narrow down the exact interference range also average packet delivery ratio and packet delay is observed as shown in figure 5.5 and figure 5.6. The experiment is continued by halving the interval until the ideal link performance between n_1 and n_2 is reached at $d = 291$ metres.

Notice the packet delay is more sensitive than throughput and packet delivery ratio to variations of d. To improve readability a log_2 scale is applied to the vertical axis for packet delay. As a result we determine R_I 3 hops, if one hop is set to 100 metres.

Table 5.3 shows general simulation parameters, which are used for all experiments. Specific parameters for each experiment are listed in the section of each experiment.

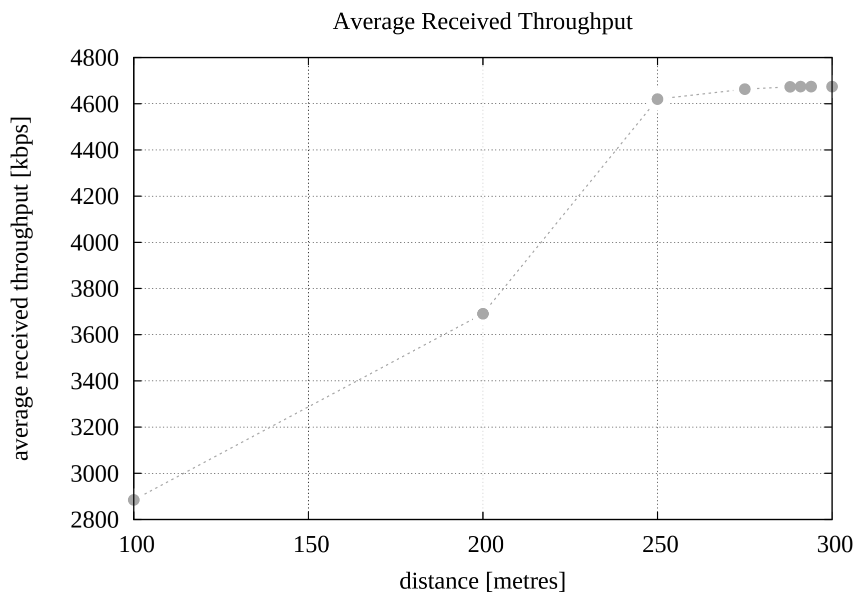


Figure 5.4: Average received throughput at node n_2

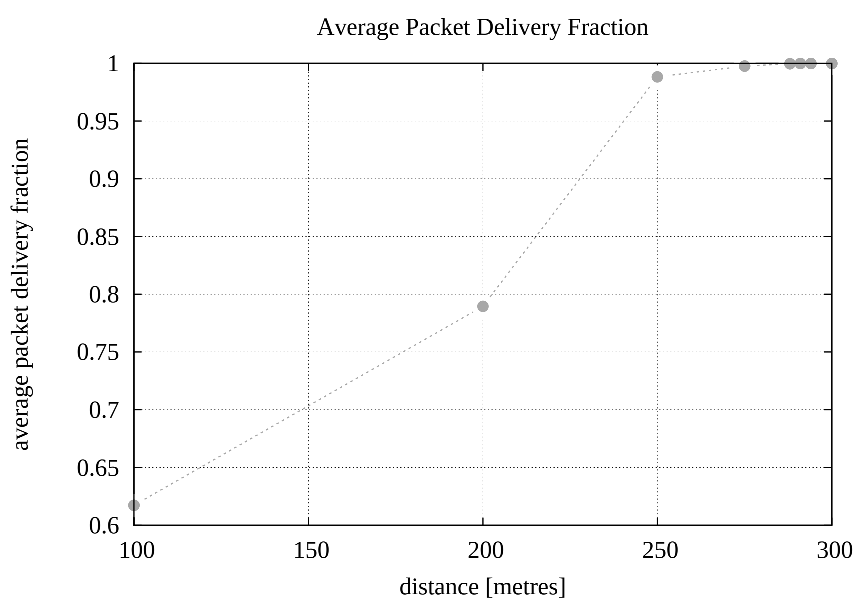


Figure 5.5: Average packet delivery ratio at node n_2

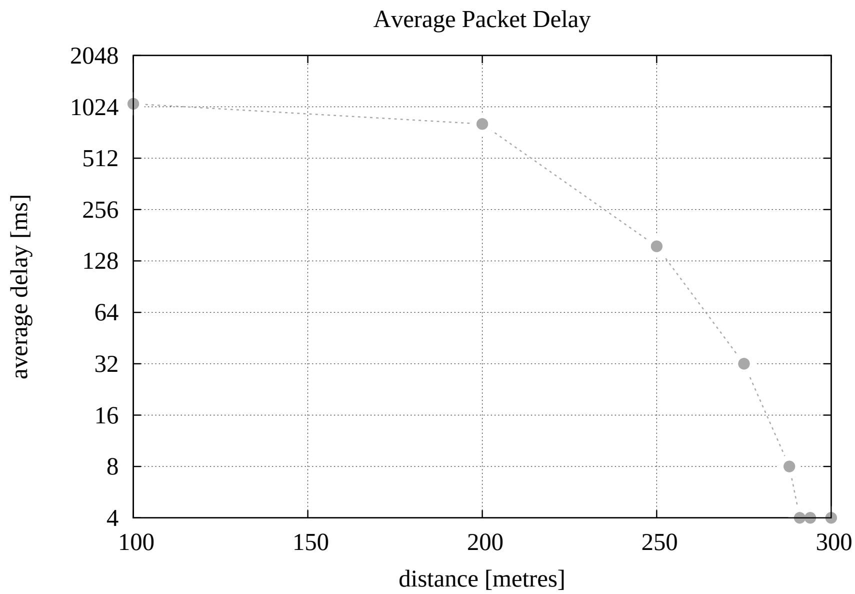


Figure 5.6: Average packet delay at node n_2

Table 5.3: General Simulation Parameters

Simulation Parameters	**Values**
Communication Range	100m
Interference Range	291m
Physical Layer	IEEE 802.11a
Mac Layer	IEEE 802.11s
Signal Propagation Model	Constant Speed of Light
Signal Loss Model	Logarithmic Distance (47dB at 1m)
Error Rate Model	Nist [64]
Gross Data Rate	dynamic, at 100m: 12 Mbps
Net Data Rate	dynamic, at 100m: 4.5 (measured) Mbps
Traffic Model	UDP/CBR
Traffic Application	OnOff
Data Packet Size	1024 bytes

5.2.2 Evaluation Metrics

Two kinds of evaluation metrics are used. The first kind are per flow metrics meaning a flow monitor [16] observes various parameters between source and sink of UDP traffic and delivers figures of merit for that flow. The second kind are per node metrics, irrespective of direct participation in a UDP traffic flow. Initially the per flow metrics are defined (THP, PDF, DELAY), eventually per node metrics (RLR, QUEUE, PREQ, PREP, PERR, CLOSED, OPENED).

Flow throughput (THP) is defined in equation 5.1 as the sum of all bits received b_r from first bit to I^{th} bit over the difference between arrival time of the first packet t_f and arrival time of the last packet t_l.

$$THP = \frac{\sum_{i=1}^{I} b_r}{(t_f - t_l)} \tag{5.1}$$

Packet Delivery Fraction (PDF) is defined in equation 5.2 as the number of packets received p_r (first to I^{th} packet) divided by the number of packets transmitted p_t (first to J^{th} packet).

$$PDF = \frac{\sum_{i=1}^{I} p_r}{\sum_{j=1}^{J} p_t} \tag{5.2}$$

End-to-End Delay (DELAY) is defined in equation 5.3 as the sum of delay of all received packets d_r divided by their number p_r (first to I^{th} packet).

$$DELAY = \frac{\sum_{i=1}^{I} d_r}{\sum_{i=1}^{I} p_r} \tag{5.3}$$

All flow metrics are averaged over the number of flows (first to F^{th} flow) according to equation 5.4.

$$FlowAverage = \frac{\sum_{f=1}^{F} metric}{F} \tag{5.4}$$

The following metrics are calculated by observed simulation parameters on individual nodes, irrespective if the node is a UDP source, sink or a passive (potentially forwarding) node.

Routing Load Ratio (RLR) is defined in equation 5.5 as the number of routing bytes transmitted rb_r (first to I^{th} byte) divided by the number of data bytes received db_r (first to J^{th} byte). The transmitted routing bytes are counted on all nodes. The received data bytes are counted for all flows. The same is valid for the RLR in terms of packets.

$$RLR = \frac{\sum_{i=1}^{I} rb_t}{\sum_{j=1}^{J} db_r} \tag{5.5}$$

Queue Length is defined in equation 5.6 as the number of packets in Enhanced Distributed Channel Access (EDCA) transmission queue measured every $t = 10ms$ averaged over the number of measurements M, summed for all interfaces I on a node. The NS-3 implementation of 802.11s has multiple queues at the link layer for different kinds of frames. We found that the EDCA queue is the only one that correlates positively with the traffic load. The queue has a size of 400 frames. We suppose that if a routing protocol can setup a path better with regard to IaFI and channel metric than another, the queue should contain less data packets on average, because nodes have to wait less to access the the channel. This metric expresses how busy on average all EDCA transmission queues are on all nodes of the network.

$$QUEUE = \sum_{i=1}^{I} \left(\frac{\sum_{m=1}^{M} queueLength_t}{M}\right) \tag{5.6}$$

The same principle as in equation 5.6 is applied to average number of received PREQ, PREP, Path Error Packet (PERR) frames and average number of links opened and links closed by the Peer Link Protocol. The per node metrics are then averaged over the number of all nodes N in the network.

5.2.3 Chain Network Performance

In this section we compare the protocols and metrics in a chain topology with one flow. Because only one path exists in a chain topology, it is well suited to compare routing protocols which are supposed to exploit channel diversity and minimize IaFI.

We set the transmission bit rate to 4550 Kbps, this distance on a single hop 100% packet delivery is achieved. Traffic source and sink are at opposite sides of the chain. Every node has 3 radio interfaces with 3 orthogonal channels. Simulation time is set to 300 seconds. Source and sink change their position to the opposite side for every other run. The results show an average of 10 runs, which is enough for a representative evaluation considering the stable circumstances in a chain with a single flow. We do not consider standard deviation or confidence intervals because there is negligible variation in the performance because of the chain topology. The channel metric for each link in the chain is achieved with an estimation of ALM based on channel bandwidth, channel access overhead and average frame error rate as specified in 802.11s standard [88]. Table 5.4 lists the experiment specific parameters, for general parameters see table 5.3.

Figure 5.7 shows the average received throughput as the chain length increases. Because HWMP with ALM is blind to channel diversity it shows the lowest throughput as expected. Also conform with our expectations is the advantage of HWMP with

Table 5.4: Simulation Parameters

Simulation Parameters	**Values**
Chain Length	2-20 nodes
Simulation Area	chain with 100m steps
Number of Interfaces/Channels	3
Routing Protocol	HWMP/RDV
Destination Only Flag	true
Data Rate	4550 Kbps
Source, Destination Allocation	opposite of chain
Number of Flows	1
Simulation Time	300 s
Simulation Runs	10 / chain length

WCETT for short paths, but little to no advantage for longer paths. The reason for this is that WCETT uses a so called pessimistic [23] IaFI model, where it assumes that all links on the same channel within a path interfere with each other, which is not true for links that are far apart. The result is an even balancing of the channels across the path, which can result in consecutive channel uniform hops on long paths. As can be seen HWMP and our proposed metric CIETT offers more throughput because it prefers paths where the last two channels have not already been used.

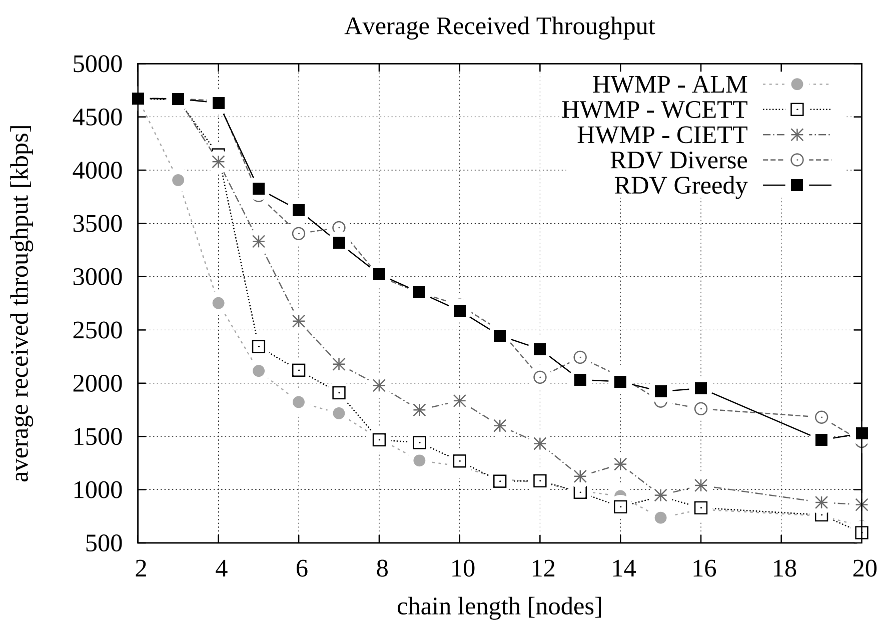


Figure 5.7: Average received throughput in a chain.

However the large discrepancy between RDV and HWMP with CIETT demonstrates the overall improvement that RDV offers with its new path formation mechanism over a classical AODV based path formation. One can see that Diverse is sometimes slightly better than Greedy at chain length 7, 13 and 19. There is no particular reason for this, since because no concurrent flows exist in this scenario which could influence channel metrics, both algorithms should perform the same, since both create a diverse channel sequence. We assume this is due to other factors in protocol functionality which might be influenced by the rigid channel selection scheme $(A, B, C, A, B...)$ of the Diverse algorithm.

Figure 5.8 shows average end-to-end packet delay. The relative performance as seen in throughput remains.

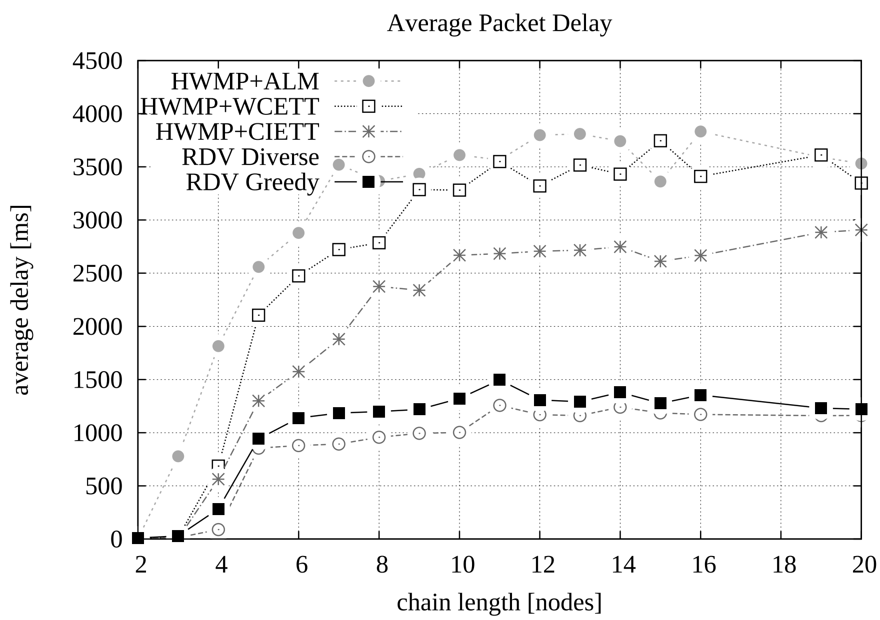


Figure 5.8: Average packet delay in a chain.

Interestingly the Diverse channel selection algorithm offers the lowest delay. The reason might be that the Diverse algorithm due to its lower complexity compared to Greedy is executed quicker and reduces packet delay during the path formation phase. However we could not validate that the algorithm runtime is realistically modelled in NS-3. The life time for a path in our simulation environment is 5 seconds, which means that the path formation mechanism is repeated cyclically after that period. We investigated the issue by artificially prolonging the path lifetime to e.g. 50 seconds, but that did not change the outcome. It is therefore unclear to us why Diverse shows a lower delay in that particular chain scenario, but the reason should be an interesting detail, which could help to improve the protocol further. We add this fact to the list of future work items found in appendix A.2.

What must be noted is that in both figures one can see that at a chain length of 4 or more all protocols steeply decrease performance wise, even RDV, which is reliably channel diverse. We account this behaviour to the fact, that our assumption that R_I id 3 times of communication range is not quite true in practice. It seems that the first and the third hop still create IaFI, possibly through an addition of noise from links that use the same channel three hops away in both directions. Notice that with a 3 channel network this is the minimal IaFI possible. More channel diversity and less IaFI can only be achieved with more radio interfaces and orthogonal channels. If Common Channel Assignment (CCA) is to be maintained the amount of non overlapping channels must be equal to the number of radio interfaces. More orthogonal channels than radio interfaces would require Varying Channel Assignment (VCA), which is not assumed in our system model, see section 3.1. For practicability reasons we do not increase the amount of radio interfaces and keep the setup at three interfaces for other experiments.

5.2.4 Grid Network Performance

In this section we compare the protocols and metrics in a grid topology with 16 flows. The grid contains 64 nodes and with 16 flows, half of the nodes are either source or sink, while the other half are potential intermediary nodes. Because the source/sink pairs are chosen randomly, the experiment is repeated 50 times. To test the performance under varying loads, the bit rate is increased from 100 to 1200 Kbps. The experiment specific simulation parameters are summarized in table 5.5, for general parameters see table 5.3.

Table 5.5: Simulation Parameters

Simulation Parameters	**Values**
Node Population	64
Simulation Area	grid with 100m step
Number of Interfaces/Channels	3
Routing Protocol	HWMP/RDV
Destination Only Flag	true
Data Rate	100-1200 Kbps
Source, Destination Allocation	random
Simultaneous Flows	16
Simulation Time	100 s
Simulation Runs	50 / data rate

We report average received throughput, average packet delay, average packet delivery fraction and average routing load ratio in packets and bytes respectively. To better understand the above metrics we also analyse average transmission queue length, received PREQ, PREP, PERR frames and links opened and closed by the PMP. Throughput, delay and delivery fraction are an average over all flows per run and over all runs. Routing load ratio is defined as the number of data packets received over the number of management frames transmitted. For WCETT we use β of 0.5. Figure 5.9 shows the average received throughput. Similar to the chain experiment one can see the relative advantage of RDV over HWMP. It is curious to observe that Diverse and Greedy channel selection perform the same until the per flow load is increased to 400 Kbps. We assume that with low bit rate flows, the distribution of channel metrics resembles the Poisson distribution function. It presumably changes with higher bit rates towards a more uniform distribution as links that are used by multiple flows suffer more from interference which increases the error rate function and raises ALM.

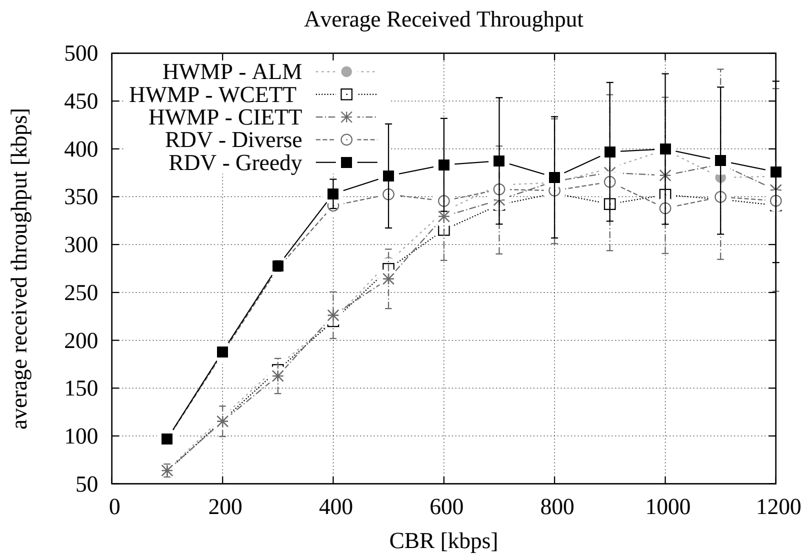


Figure 5.9: Average received throughput.

What is fundamentally different to the chain experiment, is that HWMP with IaFI aware metrics, such as WCETT and CIETT does not perform better than with ALM. This observation underlines the advantage of RDV over approaches that aim at substitution of channel metrics. Further we only plot error bars for RDV - Greedy and HWMP - CIETT for readability reasons. What can be observed is that RDV has a lower standard deviation for lower network loads, which we interpret as a more consistent performance behaviour for that case. Notice that the range of 100 to 300 Kbps the deviation of RDV is too small to be visible on the plot. Also remarkable is the fact that under high load of 800 to 1200 Kbps all protocols perform similar. An interesting question is why RDV is beneficial under lower load but does not to distinct itself from HWMP under high load? We assume that under high load the available channel bandwidth within the network is exhausted through network traffic saturation. Under saturation packet flows use all available bandwidth and compete with each other for channel access, they create IrFI. We know from our chain network experiment that RDV creates less IaFI than HWMP, because this scenario has no IrFI, thus we assume that in case of network saturation IrFI increases to such an extent that it negates the benefits of RDV. But also notice that even under saturation RDV on average performs better than HWMP.

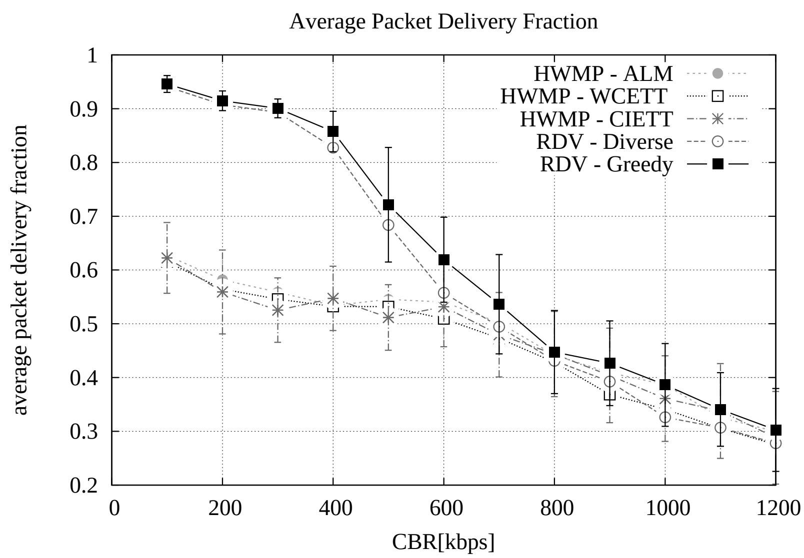


Figure 5.10: Average received packet delivery fraction.

Figure 5.10 shows the average packet delivery as a fraction of total packets sent. The overall performance relation remains as with average received throughput. It shows that for example RDV achieves a 65% packet delivery rate at about 500 Kbps, while HWMP manages this rate at 100 Kbps. Figure 5.11 shows the average end-to-end packet delay. This result is a curiosity. The relative advantage of RDV as seen in the chain experiment is not seen in the grid experiment. RDV shows a higher delay compared to HWMP for low bit rates, but a lower delay at high bit rates. However RDVs advantage is the low bit rate traffic range. The fact that Greedy performs better than Diverse is consistent with our expectations, however more investigations are necessary to determine the exact reason for the worse behaviour of RDV in terms of delay compared to HWMP. We know that the path formation mechanism of RDV, which delays the reverse path formation from the PREQ phase to the PREP phase and has a higher runtime than HWMP, is not the reason by artificially increasing the path lifetime, which showed no decrease in packet delay for RDV. It is possible that RDV prefers longer, less congested routes, with less IaFI to HWMP, however this needs to be investigated further, see list of future work items in appendix A.2.

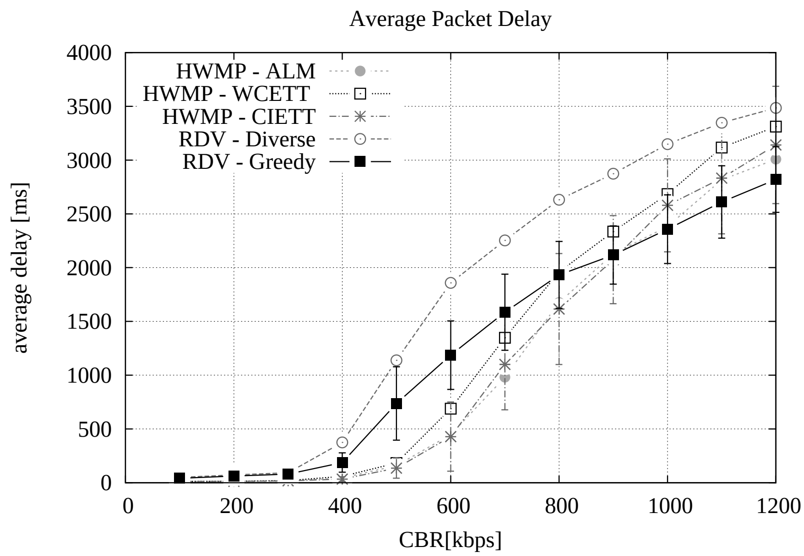


Figure 5.11: Average packet delay.

Figure 5.12 shows that RDV has less than half RLR as HWMP in terms of management frames sent over data packets received. As mentioned in section 4.1 we manage the routing load by reducing the number of PREQ frame rebroadcasts. Also the low (barely visible) standard deviation of RDV points to a consistent routing load.

Figure 5.13 shows the routing load in terms of bytes. Because RDV as well as WCETT collect a route record at every hop, which size depends on the path length, we expect a higher load compared to HWMP with ALM, which only stores a cumulative metric value in one field of the PREQ frame.

The expectation is met by WCETT, but for RDV the low routing load in terms of packets compensates the increased packet size. In general one can observe that RDV offers improvement in network throughput and packet delivery at no additional routing load.

Figure 5.14 shows the queue length of packets waiting for transmission in the EDCA [9] queue. We have found that this queue is used for transmission of data packets in the ns3 implementation with a best effort prioritization. The queue has a maximum size of 400 packets. The average queue length remains low (around zero) until a traffic load of 400 Kbps for both protocols. However it increases at loads of 500 Kbps and higher. Note that the queue length of RDV is greater than of HWMP except for high loads of 1000 Kbps and above. We expect the queue length to correlate with the busyness of a channel, meaning the more contention (as a result of IrFI

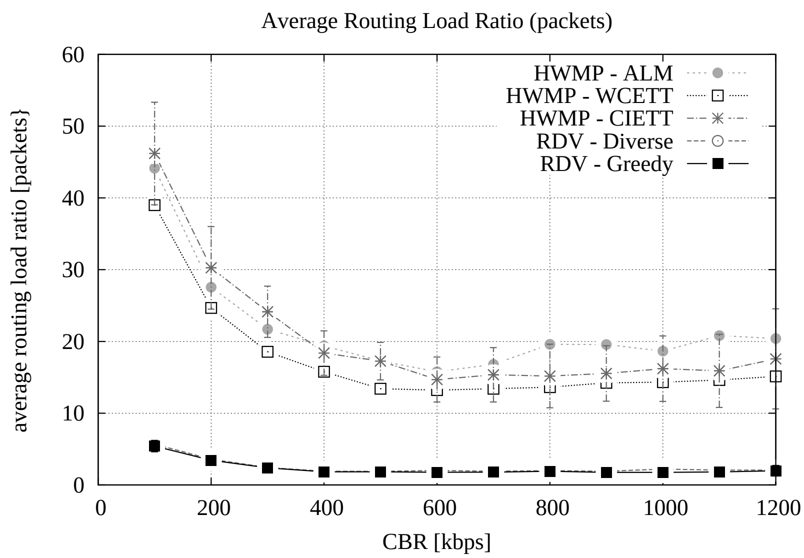


Figure 5.12: Average routing load ratio in packets.

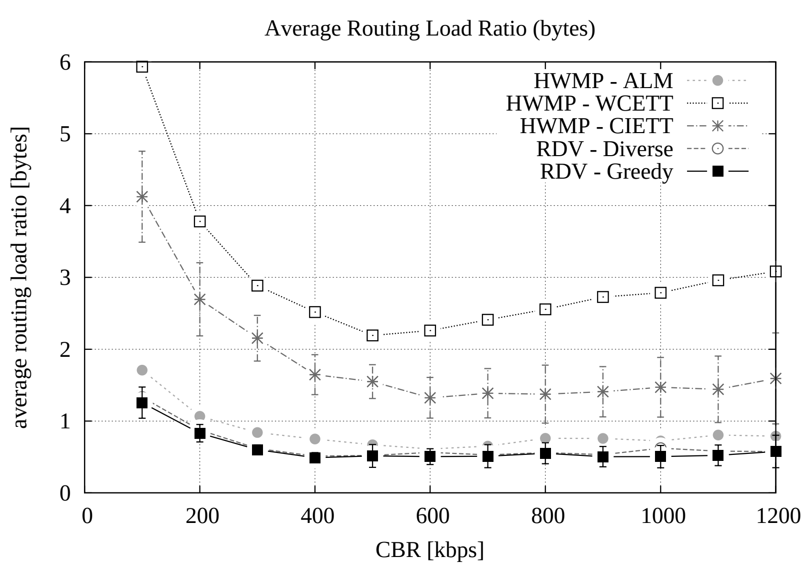


Figure 5.13: Average RLR Bytes

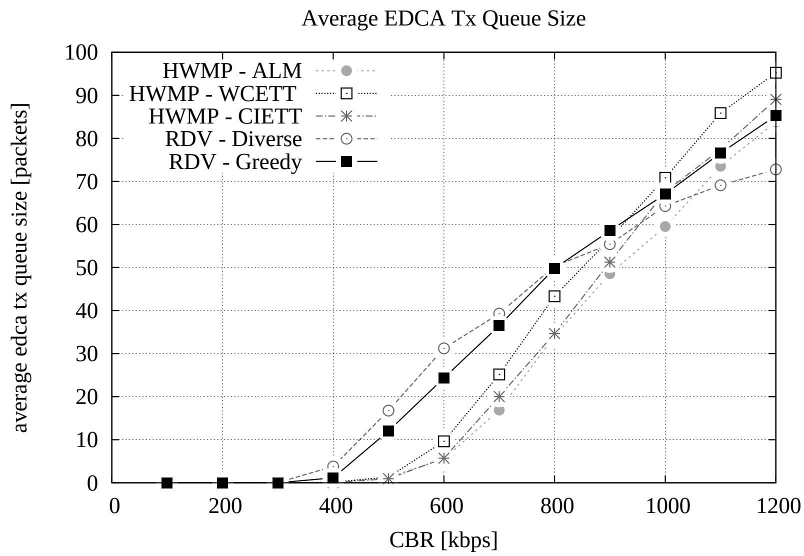


Figure 5.14: Average EDCA Transmission Queue Length

and IaFI) happens on a channel, the more packets should be waiting in queue. The interpretation of queue length with regard to performance of RDV versus HWMP is unclear yet and shall be noted for future analysis. Especially we are interested to find out the relationship of average EDCA queue length and average packet delay.

Routing load ratio considers the amount and volume of management packets sent and puts it into relation to the amount and volume of data packets received. The class of management frames consists of routing frames such as PREQ, PREP and PERR frames and peer link management frames such as peerLinkOpen, peerLinkConfirm and peerLinkClose frames used by the PMP. To better understand the differences in routing load and routing protocol functionality, we take a closer look at the average amount of those two types of frames. The basic difference between RDV and HWMP is how PREQ frames are disseminated. HWMP rebroadcasts a PREQ frame (if accepted) on all channels, which means that the amount of PREQ frames sent 'explodes' within the network. Figure 5.15 shows the average amount of received PREQ frames per node.

In fact the figure shows that RDV maintains a constant level of less than 5000 received frames on average, while HWMP varies from 15000 to 50000 frames. Interestingly the amount of received PREQ frames increases with higher bit rates for HWMP, but stays constant for RDV. Also note that the application of WCETT and CIETT reduces the number of PREQ frames compared to ALM. The reason could be the awareness of those two metrics of IaFI, so that less frames are accepted which carry a uniform channel sequence. But because both metrics rely on a route record for their

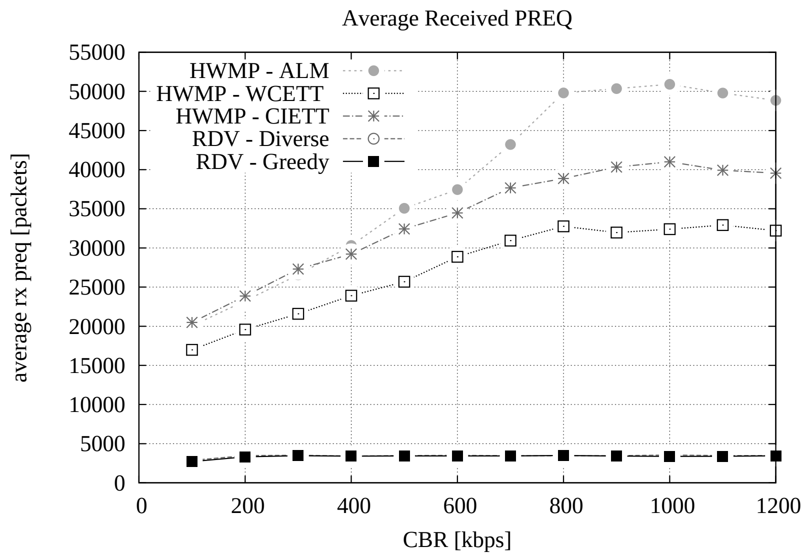


Figure 5.15: Average Received PREQ Frames

computation, the frame size increases, hence they perform considerably worse in terms of routing load volume (bytes). Overall the advantage of RDV over HWMP in terms of PREQ overhead is 1:3 to 1:10, depending on traffic load.

Whenever a PREQ frame is received and accepted at the destination node, a PREP frame is generated and sent back to the source node. Figure 5.15 depicts the average amount of received PREP frames per node in the network.

The general advantage of RDV over HWMP in terms of PREP frame overhead is 1:2 to 1:8 depending on traffic load. Because PREP frames are sent as unicast, their amount is 50 to 60 times less compared to PREQ frames.

A PERR frame is generated if a node receives a data packet for forwarding, but has no valid route to its destination node. The reason for route invalidation can be a broken link for the next hop or a route expiration. The PERR frame is used to inform the node that forwarded the data packet to the current node, that the route is lost. This procedure reduces the amount of lost data packets due to route invalidation. The forwarding node then stops the transmission of further data packets of the flow and informs its predecessor with a PERR frames, so that eventually a new route discovery process can be initialized by the source node of the flow. Figure 5.17 shows the average amount of received PERR frames per node.

It is curious to observe that RDV produces 50% to 100% more PERR messages compared to HWMP. Also the Greedy channel selection algorithm shows a slightly

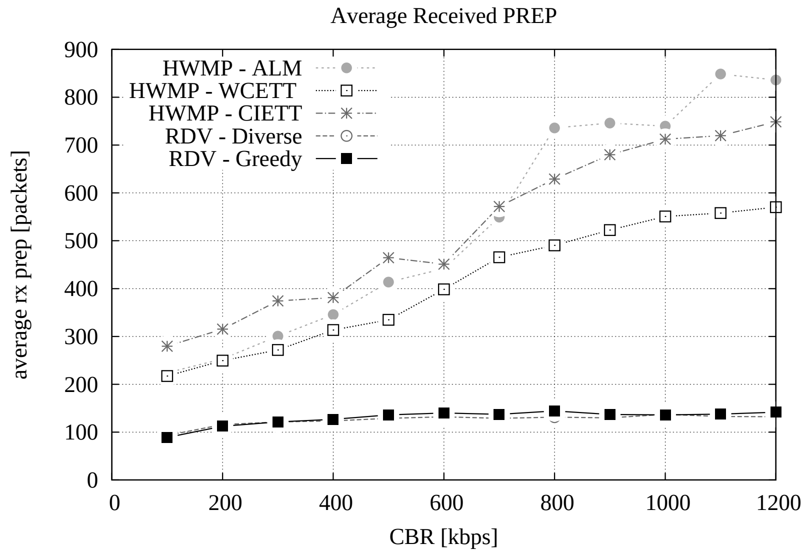


Figure 5.16: Average Received PREP Frames

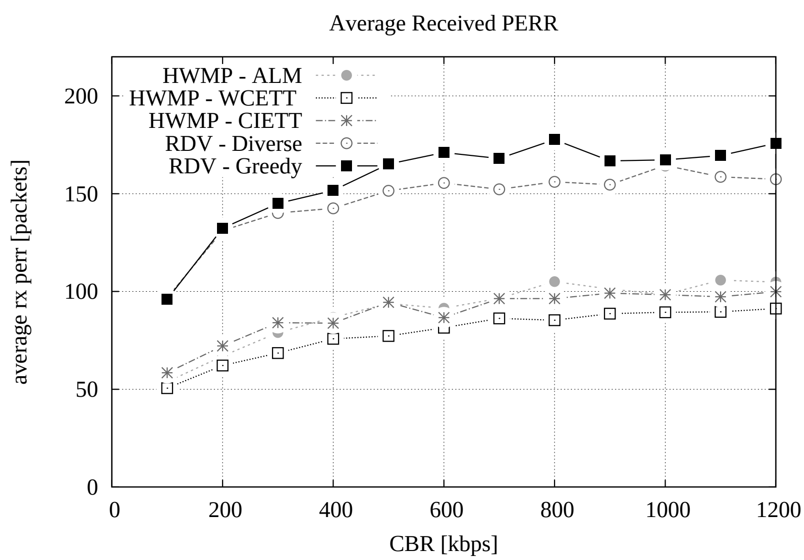


Figure 5.17: Average Received PERR Frames

higher amount compared to the Diverse channel selection algorithm. The reason for this is unclear and will be discussed in section 5.3.

Figure 5.18 and figure 5.19 shows the average amount of links opened and closed by the PMP per node. Because the holding timer of the peer link open state of the finite state machine is periodically expiring, peer links are closed and reopened on a regular base. This explains the symmetry of both plots and very similar behaviour of both RDV and HWMP. However the amount of close and open procedures also depends partially on the traffic load (100-300 Kbps). Further there is a difference between RDV and HWMP in the range of 100-800 Kbps. RDV closes more links than HWMP in the range of 100-300 Kbps and less in the range of 400-700 Kbps. This fact shall be discussed in section 5.3.

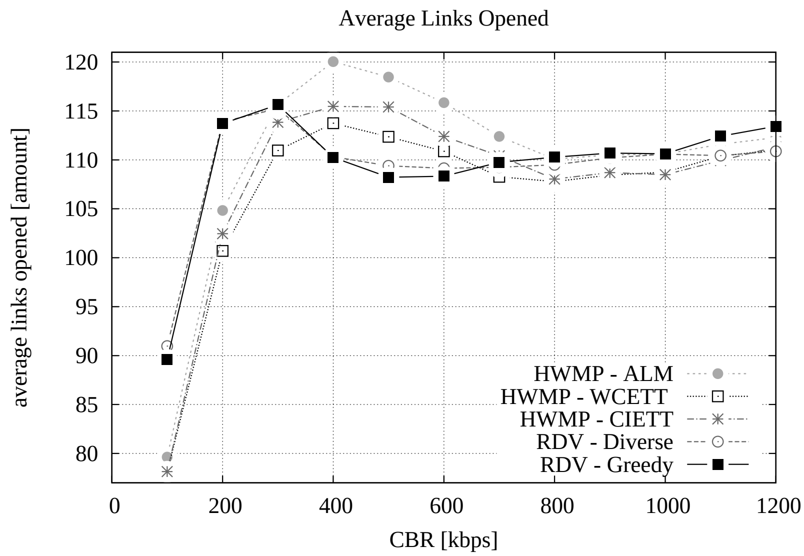


Figure 5.18: Average Links Opened

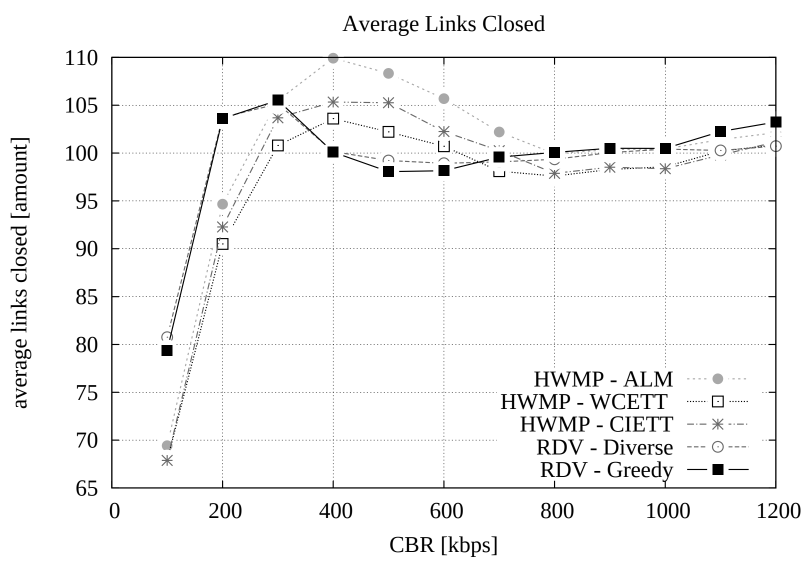


Figure 5.19: Average Links Closed

5.3 Average Packet Delay Analysis

In this section we investigate the increased average packet delay of RDV. When the data from plot 5.11 packet delay and plot 5.14 average transmission queue is observed, they seem to depend on each other.

We begin with a correlation analysis. Figure 5.20 shows a scatter plot of average packet delay for the Greedy algorithm (horizontal axis) and the corresponding average EDCA transmission queue size (vertical axis).

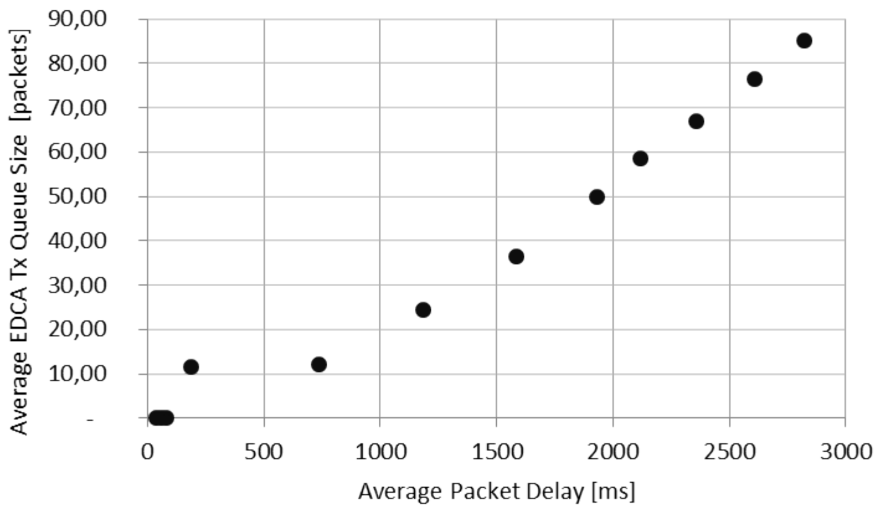


Figure 5.20: Scatter Plot Delay/Queue Length for RDV with Greedy

Indeed, the scatter plot shows that packet delay and queue size correlate with a correlation coefficient $\rho = 0.987$. Table 5.6 shows ρ for every protocol and metric combination used in the plots.

Table 5.6: Correlation Coefficients Packet Delay and EDCA Transmission Queue Size

Protocol	ρ
RDV Greedy	0.987
RDV Diverse	0.995
HWMP ALM	0.991
HWMP WCETT	0.991
HWMP CIETT	0.993

We derive from this that packet delay could be the result of packets waiting in queue to be transmitted as indicated by the strong correlations. In figure 5.11, the delay for RDV is higher than HWMP in the traffic region of 100 to 800 Kbps for Greedy and higher for all traffic loads for Diverse. A reason for increased waiting in transmission queues could be the increased PERR amount of RDV over HWMP as seen in figure 5.17. The intuition in this case is that if a PERR frame is received by a node, its neighbour node informs it that a destination of interest is no longer reachable through a previously set up path, which could lead to buffering of data packets and rediscovery of the path through the path formation mechanism. Table 5.7 list ρ for the correlation of packet delay and PERR. However the amount of average PERR frames received increases less steeply and the correlation coefficients indicate a less strong dependence of delay/queue size and PERR frame reception.

Table 5.7: ρ for Packet Delay and Received PERR Packets

Protocol	ρ
RDV Greedy	0.736
RDV Diverse	0.760
HWMP ALM	0.754
HWMP WCETT	0.813
HWMP CIETT	0.714

In fact if the increased PERR amount should be responsible for increased delay, the amount of PREQ frames should increase in the area of 100-800 Kbps, but it remains constant for both Greedy and Diverse as seen in figure 5.15. Further we have investigated the reason for PERR events in our implementation and found that two cases occur. In the first case a PERR frame is issued if no valid route is available when an intermediary node tries to forward a data packet. In the second case a PERR frame is issued if a link that is used for a path is closed. The second case is prevalent in more than 99% of the time in our experiments. Table 5.8 shows ρ for correlation of average PERR frames received and average 'LinkClosed' frames received. Again we see a rather loose correlation of link closing and PERR.

Table 5.8: ρ for PERR LinksClosed

Protocol	ρ
RDV Greedy	0.677
RDV Diverse	0.713
HWMP ALM	0.667
HWMP WCETT	0.742
HWMP CIETT	0.746

In general we can say that packet delay is indicated to be a result of queue size, but whether it is due to increased PERR is questionable. Notice correlation does not explain causalities. Also in figure 5.11, Diverse has the highest delay for all traffic loads, but its queue length varies and is lowest in the region of 1100-1200 Kbps. All in all we find the evidence for a reasoning for packet delay of RDV inconclusive and future research should be conducted as recorded in section A.2.

5.4 Chapter Summary

In this chapter we have analysed the Greedy and Diverse channel selection algorithms numerically using the Matlab environment and found that they can deliver useful optimization results for the channel selection problem stated in section 4.2 with relatively low time and memory demand.
Further we have gathered empirical evidence through simulation with the NS-3 environment to support our hypothesis, that the RDV concept can reduce IaFI better than HWMP and improve network performance except for packet delay in lower traffic load scenarios. In the following chapter we will conclude our work and discuss future work.

Chapter 6

Conclusion

Contents

6.1 Summary of Results

The study of the field of research found that a multitude of previous work has been conducted on understanding and improving the performance limitations of wireless multi-hop networks. The main limitations for network size at useful performance levels have been identified as interference from neighbouring flows termed inter-flow interference and from the flow itself, when adjacent hops reuse the same channel termed Intra Flow Interference (IaFI). Our original motivation for this work is to improve wireless multi-hop network performance by reducing IaFI. To mitigate those limitations multiple radio interfaces with orthogonal channels are introduced to wireless multi-hop networks, which create complex problems, such as channel assignment and channel aware routing. We have chosen the most simple and practical form of channel assignment with a set of common channels Common Channel Assignment (CCA) on all participating nodes. Following this we have found that exploiting channel diversity of CCA multi-hop networks is a challenge if a distance vector based routing protocol is used. Consequently we focus on channel aware routing in the presence of multiple available links between two nodes and its challenges. With our chosen platform, IEEE 802.11s, we have found that a classic distance vector based routing protocol based on AODV such as HWMP, when combined with IaFI aware routing metrics, offers very little to no benefit if CCA is used, when analysed experimentally.

To prove that we have implemented the WCETT routing metric, which is designed to prefer paths with diverse channel sequences. Because WCETT is blind to the distance between channel uniform hops, it performs better then the original interference ignorant metric ALM, but only on short paths. To solve that problem we contribute to the field of research with a simple IaFI aware routing metric termed

Cumulative Interference aware Expected Transmission Time (CIETT). CIETT offers minor improvements also for longer paths compared to WCETT over ALM. Because CIETT's approach is similar to other concepts of context aware routing metrics in literature, which tackle the problem with yet another (more sophisticated) interference aware routing metric calculation, we understand that this concepts are not enough to significantly decrease IaFI and improve network performance.

Through analysis of distance vector path formation procedures with different routing metrics (ALM, WCETT, SIM and CIETT) in chapter 3 we have found that the reason for disappointing performance improvement might not lie in the metric calculation, but also in the path discovery mechanism if CCA is used. Because a distance vector routing protocol distributes the optimization decisions to intermediate nodes and because those nodes have no knowledge of the rest of the path and available channels (to the destination node), they exclude potentially valuable links with certain channels from the final channel sequence between source and destination.

Hence we identify the path formation mechanism of HWMP and therefore AODV as the main culprit for IaFI minimization as the reason for a lack of significant performance improvement under our system model. Drawing from this knowledge we redesign the distance vector path formation mechanism. Our design collects a route record along all intermediary nodes between a flow source and destination and relocates the optimization decision to the destination node, where path knowledge is complete. Our design maintains the distance vector principle as described in chapter 4.

We state our hypothesis that our new routing protocol Route record based Distance Vector (RDV) is better suited to minimize IaFI, hence improves performance better than a classic distance vector protocol, in this case HWMP. However RDV poses a new challenge of time and memory expensive discovery of an optimal channel sequence between source and destination as described in section 4.2. To thoroughly understand the problem, we theoretically analyse it in section 4.3 and find that the problem can be reduced to a classical sequence labelling problem from the domain of graph theory. Further it can be approximately represented by the Markov property. We use this property to reduce the complexity to polynomial time and memory demand. Following we apply the Viterbi algorithm, a procedure from the domain of dynamic programming usually used to efficiently solve complex problems in applications such as speech recognition, to find an optimal solution in a comprehensive example assuming a second order Markov chain. We design a simplistic channel sequence selection algorithm, which irrespective of link quality and contention, very roughly optimizes the channel sequence at negligible time and memory cost, termed Diverse channel selection algorithm described. Also we design an approximation to the Viterbi algorithm, which demands less time and memory cost, the Greedy channel selection algorithm as described in section 4.5. We provide an analysis for the complexity and the approximation error of both algorithms.

Finally in chapter 5 empirical evidence is gathered. A numerical analysis of both approximations in relation to the Viterbi algorithm is presented in section 5.1. We find that they can deliver viable optimization results, depending on the traffic profile. In section 5.2 we present evidence regarding our stated hypothesis from extensive discrete event simulations. The results show that RDV, when applied with Diverse and Greedy channel selection algorithms significantly improves packet delivery and per flow throughput but at the cost of increased packet delay for some traffic loads, a feature that demands further study. We account this improvement to RDV's ability to establish distance vector paths with channel sequences with minimal IaFI. The evidence also shows that the routing load volume of RDV is on par or below of classic distance vector protocols represented by HWMP. Please note that although our assumed CCA scheme may be limiting to the application area of RDV, our protocol is also applicable to VCA schemes, if some nodes share more than one common channel.

The contributions of this work can be summarized as follows:

- As first contribution we analyse and identify the weakness of HWMPs path formation mechanism with regard to Forward Path Unawareness (FPU).
- As second contribution we propose our redesigned routing protocol RDV as a remedy to the path formation problem of HWMP. However RDV poses a new challenge, the channel sequence selection problem.
- As third contribution we provide the theoretical analysis of the channel sequence selection problem and a solution, the Viterbi channel selection algorithm and its practical approximation the Greedy channel selection algorithm.
- As fourth contribution we provide an analysis of our protocol and algorithms through numerical model evaluation and simulation.

In chapter 5 we present the numerical and simulative evaluation of RDV with supportive evidence to our stated hypothesis. The state of the research field in light out our contributions is extended by a new approach to distance vector routing for multichannel architectures with the ability to minimize IaFI. Further it is complemented by an adaptation of a second order Markov-Viterbi algorithm.

6.2 Future Work

For further research this work suggests an investigation of the packet delay phenomenon of RDV compared to HWMP. The observation that RDV generates more

PERR events and a longer transmission queue demands investigation, especially with regard to packet delay and possible further performance improvements.

Also implementation and evaluation of the Viterbi algorithm and its implications to computational efficiency compared to the Greedy algorithm needs consideration. Further it can make sense to use all three channel selection algorithms Diverse, Greedy, Viterbi on the destination node in a dynamic manner depending on available computation and memory resources at time of PREQ packet reception at the destination node. If resources are abundant, the Viterbi algorithm can be used primarily, under mid range load Greedy can offer fair optimization results and under high load the Diverse channel selection algorithm can deliver the minimum of optimization. Also a parallelization of the Viterbi algorithm on multi core processing architectures can create a benefit and can be analysed eventually, as well as the impact of more than three interfaces and channels on algorithm performance.

A further study of other channel metrics with respect to their distribution function to better exploit the ability of RDV to optimize both for channel diversity and channel busyness. Another future work is the extension of the RDV principle to the proactive mode of HWMP. In this mode a so called root node pro-actively broadcasts PREQ packets to all nodes of the network in order to establish a path towards the root node on all nodes. Because this mode is also distance vector based an adaptation should be possible with moderate effort.

Network coding such as studied in [47] can be used to reduce the amount of retransmissions of lost packets due to interference. The idea is to store overheard packets temporarily on nodes and in case of packet loss use the XOR operator on retransmitted frames of multiple packets by multiple sources. However the mentioned work is focused on single hop networks where a central entity (access point) can coordinate network coded retransmissions. The lack of such an central entity in multi-hop networks poses a challenge for coordination and presents an interesting research field. The work of [32] also falls into this category.

Eventually this protocol should be evaluated in a test-bed for a more comprehensive study under real conditions.

Appendix A

Appendix

A.1 NS-3 Class Diagram

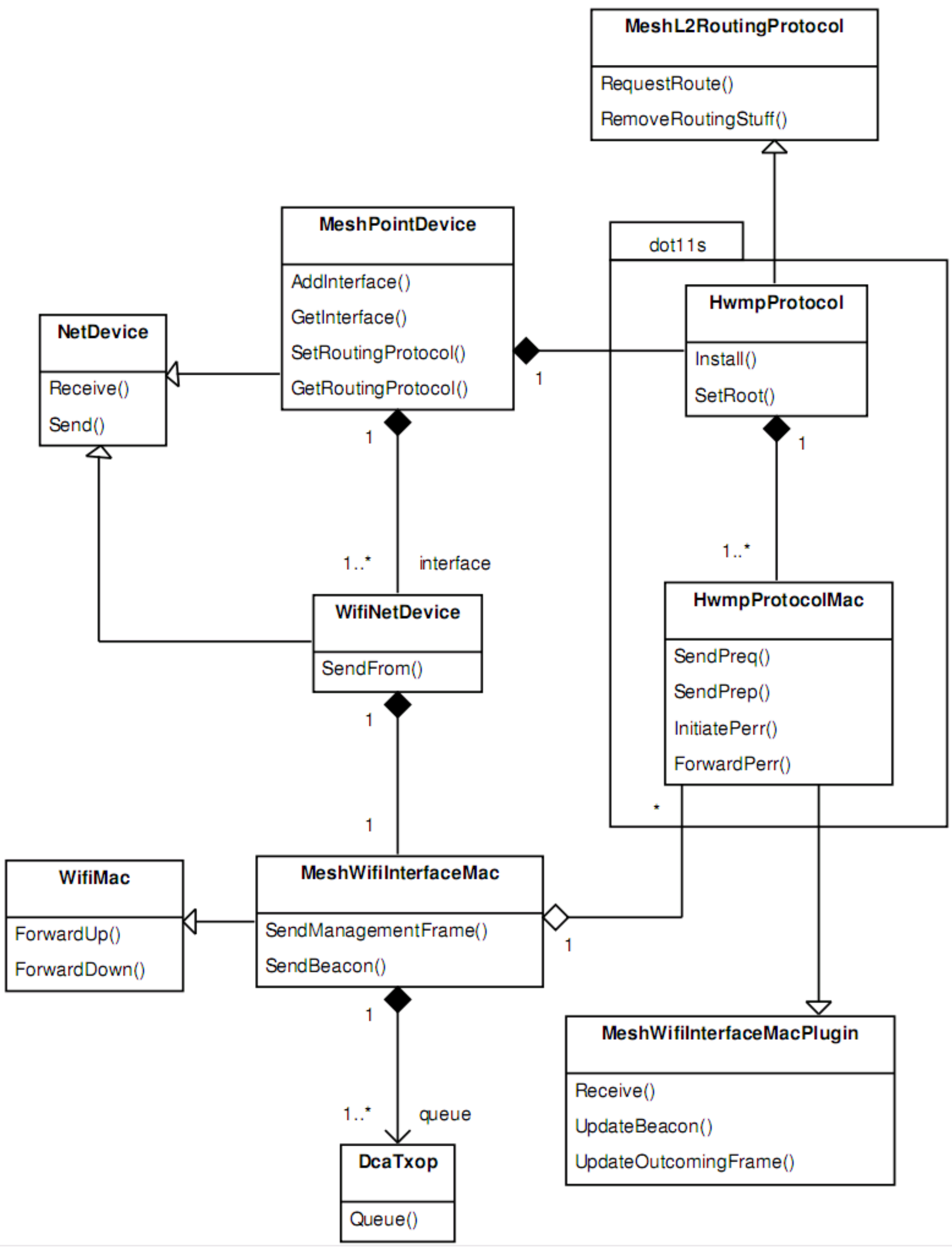


Figure A.1: IEEE 802.11s NS-3 Class Diagram

A.2 List of Future Work Items

Table A.1: Future Work Items

Topic	Item
Numerical Evaluation section 5.1	1. Evaluation of channel selection algorithms with other topologies (grid, random, mobility) and channel metric distributions. Plotting and interpretation of standard deviation and estimation of runtime for the algorithms.
Simulative Evaluation section 5.2.4	1. Different measurement of link layer queue, to reflect long term queue status better, possibly Exponentially Weighted Moving Average. 2. Investigation of reason for why Diverse has a lower delay than Greedy in figure 5.8. Addition of confidence intervals. Initial investigations show that path formation delay plays no role, also algorithm runtime in NS-3 is not accounted for. 3. In grid network experiment, the data used in the plots represents average values of flows with heterogeneous length. It is necessary to analyse the scenario in such a way that average values of flows with homogeneous length is represented. The goal of this analysis is to find out if RDV is more beneficial to short or long flows. 4. Determination of average path length for RDV and HWMP for grid network experiment. If the average path length of RDV is longer than HWMP this could explain its higher delay in case of moderate traffic load. Notice that in NS-3 flow monitor is used to report per flow statistics. However flow monitor is IP based and from its perspective the number of times a packet is forwarded in a 802.11s network is 0.

Bibliography

[1] A. Adya, P. Bahl, J. Padhye, and A. Wolman. A multi-radio unification protocol for IEEE 802.11 wireless networks. In *First International Conference on Broadband Networks*, pages 344–354. IEEE Comput. Soc, 2004. (Cited on pages 16 and 20.)

[2] M. Alicherry, R. Bhatia, and L.E. Li. Joint Channel Assignment and Routing for Throughput Optimization in Multiradio Wireless Mesh Networks. *IEEE Journal on Selected Areas in Communications*, 24(11):1960–1971, November 2005. (Cited on page 16.)

[3] JB Andersen. Propagation measurements and models for wireless communications channels. *Communications Magazine, IEEE*, 33(1):42–49, 1995. (Cited on page 8.)

[4] K Andreev and P Boyko. IEEE 802.11 s Mesh Networking NS-3 Model. In *Workshop on ns3*, 2010. (Cited on pages 43 and 65.)

[5] Stefano Avallone and Ian F. Akyildiz. A Channel Assignment Algorithm for Multi-Radio Wireless Mesh Networks. In *2007 16th International Conference on Computer Communications and Networks*, pages 1034–1039. IEEE, August 2007. (Cited on page 16.)

[6] Paramvir Bahl, Ranveer Chandra, and John Dunagan. SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking - MobiCom '04*, page 216, New York, New York, USA, September 2004. ACM Press. (Cited on pages 14 and 15.)

[7] Nicholas D Bamblos, Shou C S.C. Chen, and G.J. Gregory J Pottie. Radio link admission algorithms for wireless networks with power control and active link quality protection. In *Proceedings of INFOCOM'95*, pages 97–104. IEEE Comput. Soc. Press, 1995. (Cited on page 9.)

[8] R Bellman. On a routing problem. *Quarterly of Applied Mathematics*, 16:87–90, 1956. (Cited on page 24.)

[9] M Benveniste. 'Tiered contention multiple access'(TCMA), a QoS-based distributed MAC protocol. *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2002.*, 2:598–604, 2002. (Cited on page 78.)

[10] Amira Bezzina, Mouna Ayari, Rami Langar, and Farouk Kamoun. An interference-aware routing metric for multi-radio multi-channel wireless mesh networks. *2012 IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 284–291, October 2012. (Cited on page 26.)

[11] Rishi Bhardwaj, Chandrakanth Chereddi, Pradeep Kyasanur, and Paul Roycroft. Net-X : A Framework for Supporting Multiple Channels and Multiple Interfaces in a Wireless Network. *UIUC Technical Report*, pages 4–6, 2007. (Cited on page 16.)

[12] Albert T Bharucha-Reid. *Elements of the Theory of Markov Processes and their Applications*. Courier Dover Publications, 2012. (Cited on page 48.)

[13] A Brzezinski, G Zussman, and E Modiano. Enabling distributed throughput maximization in wireless mesh networks: a partitioning approach. *MobiCom '06 Proceedings of the 12th annual international conference on Mobile computing and networking*, 12:26–37, 2006. (Cited on page 16.)

[14] University of Southern California. Internet Protocol. *IETF*, RFC 791(Updated by 1349, 2474, 6864), 1981. (Cited on page 6.)

[15] Alberto Caprara, Hans Kellerer, Ulrich Pferschy, and David Pisinger. Approximation algorithms for knapsack problems with cardinality constraints. *European Journal of Operational Research*, 123(2):333–345, June 2000. (Cited on page 18.)

[16] Gustavo Carneiro, Pedro Fortuna, and Manuel Ricardo. FlowMonitor - a network monitoring framework for the Network Simulator 3 (NS-3). *Proceedings of the 4th International ICST Conference on Performance Evaluation Methodologies and Tools*, 3:10, 2009. (Cited on page 71.)

[17] Edward Cave. The Heliotrope a New Instrument. *The Gentleman's Magazine*, 92(2):358–359, 1822. (Cited on page 5.)

[18] VG Cerf and RE Icahn. A protocol for packet network intercommunication. *ACM SIGCOMM Computer Communication Review*, 2005. (Cited on page 6.)

[19] SO Cheikh, MM Hassan, and A Geuroui. New Metric for HWMP Pprotocol (NMH). *International Journal of Computer Networks & Communications (IJCNC)*, 5(2):49–58, 2013. (Cited on page 26.)

[20] H. Cheng, N. Xiong, G. Chen, and X. Zhuang. Channel assignment with topology preservation for multi-radio wireless mesh networks. *Journal of Communications*, 5(1):63–70, 2010. (Cited on page 16.)

[21] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol (OLSR). *IETF*, RFC 3626, 2003. (Cited on page 24.)

[22] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris. a High-Throughput Path Metric for Multi-Hop Wireless Routing. *Wireless Networks - Special issue: Selected papers from ACM MobiCom 2003*, 11(4):419–434, July 2005. (Cited on pages 21 and 25.)

[23] Saumitra Das, Yunnan Wu, Ranveer Chandra, and Y Charlie Hu. Context-based Routing : Techniques , Applications and Experience. *NSDI*, 8:379–392, 2008. (Cited on pages 25, 26, 27, 34 and 73.)

[24] S.M. Das, H. Pucha, D. Koutsonikolas, Y.C. Hu, and D. Peroulis. DMesh: Incorporating Practical Directional Antennas in Multichannel Wireless Mesh Networks. *IEEE Journal on Selected Areas in Communications*, 24(11):2028–2039, November 2006. (Cited on pages 16 and 19.)

[25] S.R. Das, C.E. Perkins, and E.M. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, volume 1, pages 3–12. IEEE, 2000. (Cited on page 10.)

[26] S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani. *Algorithms*. McGraw-Hill, 2006. (Cited on page 49.)

[27] EW Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1:269–271, 1959. (Cited on page 24.)

[28] Discrete-Event Network Simulator for Internet Systems. ns-3 - http://www.nsnam.org/, 2013. (Cited on page 65.)

[29] Richard Draves, Jitendra Padhye, and Brian Zill. Routing in multi-radio, multi-hop wireless mesh networks. *Proceedings of the 10th annual international conference on Mobile computing and networking MobiCom 04*, 1:114–128, 2004. (Cited on pages 16, 21, 25 and 26.)

[30] L.R. Ford. Network Flow Theory. *RAND Corporation*, Paper P-92:87–90, 1956. (Cited on page 24.)

[31] Haralt T. Friis. A Note on a Simple Transmission Formula. *Proceedings of the I.R.E. and Waves and Electrons*, 34:254–256, 1946. (Cited on page 7.)

[32] Shyamnath Gollakota. *Embra ing Interferen e in Wireless Systems*. PhD thesis, Massachusetts Institute of Technology, 2012. (Cited on page 92.)

[33] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, March 2000. (Cited on pages 9 and 11.)

[34] C. Hedrick. Routing Information Protocol. *IETF*, RFC 1058, 1988. (Cited on page 24.)

[35] J. Hofmueller, A. Bachmann, and I. Zmoelnig. The Transmission of IP Datagrams over the Semaphore Flag Signaling System (SFSS). *IETF*, RFC 4824, 2007. (Cited on page 5.)

[36] Ronald A Howard. DYNAMIC PROGRAMMING AND MARKOV PROCESSES. 1960. (Cited on page 48.)

[37] Kamal Jain, Jitendra Padhye, Venkata N. Padmanabhan, and Lili Qiu. Impact of Interference on Multi-Hop Wireless Network Performance. *Wireless Networks*, 11(4):471–487, July 2005. (Cited on pages 11 and 21.)

[38] N. Jain, S.R. Das, and A. Nasipuri. A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks. In *Proceedings Tenth International Conference on Computer Communications and Networks (Cat. No.01EX495)*, pages 432–439. IEEE, 2001. (Cited on page 14.)

[39] W Jiang, S Liu, Y Zhu, and Z Zhang. Optimizing routing metrics for large-scale multi-radio mesh networks. In *International Conference on Wireless Communications, Networking and Mobile Computing, WiCom 2007.*, pages 1550–1553, 2007. (Cited on page 26.)

[40] David B Johnson and David A Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. *Mobile computing*, pages 153–181, 1996. (Cited on pages 3, 10 and 21.)

[41] Jangeun Jun and M.L. Sichitiu. The nominal capacity of wireless mesh networks. *IEEE Wireless Communications*, 10(5):8–14, October 2003. (Cited on pages 1 and 11.)

[42] Vinay Kapse and Ms. U. N. Shrawankar. INTERFERENCE-AWARE CHANNEL ASSIGNMENT FOR MAXIMIZING THROUGHPUT IN WMN. *International Journal on AdHoc Networking Systems (IJANS)*, 1(1):1–12, 2011. (Cited on page 16.)

[43] Sachin Katti, M Muriel, and Jon Crowcroft. XORs in The Air : Practical Wireless Network Coding. *ACM SIGCOMM Computer Communication Review*, 36(4):234–254, 2006. (Cited on page 26.)

[44] Kernel.org. IEEE 802.11s Linux Kernel Implementation, 2013. (Cited on page 62.)

[45] Bong-Jun Ko, Dan Rubenstein, Vishal Misra, and Jitendra Padhye. Distributed Channel Assignment in Multi-Radio 802.11 Mesh Networks. In *2007 IEEE Wireless Communications and Networking Conference*, pages 3981–3986. IEEE, 2007. (Cited on pages 16 and 20.)

[46] M Kodialam and T Nandagopal. Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. *... of the 11th annual international conference ...*, 2005. (Cited on page 16.)

[47] FC Kuo, K Tan, and XY Li. XOR rescue: exploiting network coding in lossy wireless networks. *Sensor, Mesh and Ad Hoc ...*, 2009. (Cited on page 92.)

[48] Jim Kurose and Keith Ross. *Computer Networking*. Pearson, 5 edition, 2009. (Cited on page 7.)

[49] Pradeep Kyasanur, Chandrakanth Chereddi, and Nitin H Vaidya. Net-X : System eXtensions for Supporting Multiple Channels , Multiple Interfaces , and Other Interface Capabilities. *UIUC Technical Report*, August(1):1–20, 2006. (Cited on page 16.)

[50] Pradeep Kyasanur and Nitin H. Vaidya. Routing and link-layer protocols for multi-channel multi-interface ad hoc wireless networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 10(1):31–43, January 2006. (Cited on pages 16 and 19.)

[51] Jinyang Li, Charles Blake, Douglas S.J. De Couto, Hu Imm Lee, and Robert Morris. Capacity of Ad Hoc wireless networks. In *Proceedings of the 7th annual international conference on Mobile computing and networking - MobiCom '01*, pages 61–69, New York, New York, USA, July 2001. ACM Press. (Cited on page 9.)

[52] Jinyang Li, John Jannotti, Douglas S. J. De Couto, David R. Karger, and Robert Morris. A scalable location service for geographic ad hoc routing. In *Proceedings of the 6th annual international conference on Mobile computing and networking - MobiCom '00*, pages 120–130, New York, New York, USA, 2000. ACM Press. (Cited on page 10.)

[53] David A. Maltz, David B. Johnson, Yih-chun Hu, and Jorjeta Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking - MobiCom '98*, pages 85–97, New York, New York, USA, October 1998. ACM Press. (Cited on page 10.)

[54] Mahesh K. Marina, Samir R. Das, and Anand Prabhu Subramanian. A topology control approach for utilizing multiple channels in multi-radio wireless mesh networks. *Computer Networks*, 54(2):241–256, February 2010. (Cited on page 16.)

[55] JM McQuillan, I Richer, EC Rosen, and DP Bertsekas. Arpanet routing algorithm improvements. *BBN Report*, 3803(April), 1978. (Cited on page 24.)

[56] M. Medard, S.G. Finn, R.A. Barry, and R.G. Gallager. Redundant trees for preplanned recovery in arbitrary vertex-redundant or edge-redundant graphs. *IEEE/ACM Transactions on Networking*, 7(5):641–652, 1999. (Cited on page 17.)

[57] E. F. Moore. The shortest path through a maze. *Proc. Internat. Sympos. Switching Theory*, Part II:285–292, 1957. (Cited on page 24.)

[58] A. Nasipuri and S.R. Das. Multichannel CSMA with signal power-based channel selection for multihop wireless networks. In *Vehicular Technology Conference Fall 2000. IEEE VTS Fall VTC2000. 52nd Vehicular Technology Conference (Cat. No.00CH37152)*, volume 1, pages 211–218. IEEE, 2000. (Cited on page 14.)

[59] A. Nasipuri, J. Zhuang, and S.R. Das. A multichannel CSMA MAC protocol for multihop wireless networks. In *WCNC. 1999 IEEE Wireless Communications and Networking Conference (Cat. No.99TH8466)*, pages 1402–1406. IEEE, 1999. (Cited on page 14.)

[60] Anjum Naveed, Salil S. Kanhere, and Sanjay K. Jha. Topology Control and Channel Assignment in Multi-Radio Multi-Channel Wireless Mesh Networks. In *2007 IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems*, pages 1–9. IEEE, October 2007. (Cited on pages 16 and 18.)

[61] George L Nemhauser and Laurence A Wolsey. *Integer and combinatorial optimization*, volume 18. Wiley New York, 1988. (Cited on page 49.)

[62] Ns2. The Network Simulator - ns-2, 2012. (Cited on page 18.)

[63] Openwrt.org. Linux Distribution for Embedded Devices, 2013. (Cited on page 63.)

[64] Guangyu Pei and Thomas R. Henderson. Validation of OFDM error rate model in ns-3. *Boeing Research & Technology*, pages 1–15, 2010. (Cited on page 70.)

[65] CE Perkins and P Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. *SIGCOMM '94 Proceedings of the conference on Communications architectures, protocols and applications*, 24(4):234–244, 1994. (Cited on page 10.)

[66] CE Perkins and EM Royer. Ad-hoc On-Demand Distance Vector Routing. *WMCSA '99. Second IEEE Workshop on Mobile Computing Systems and Applications*, 1999. (Cited on page 28.)

[67] WW Peterson and DT Brown. Cyclic codes for error detection. *Proceedings of the IRE*, 49(1):228–235, 1961. (Cited on page 9.)

[68] J. Postel. User Datagram Protocol. *IETF*, RFC 768((Standard)), 1980. (Cited on page 6.)

[69] AHM Rad and WSV Wong. Joint Optimal Channel Assignment and Congestion Control for Multi-channel Wireless Mesh Networks. In *2006 IEEE International Conference on Communications*, pages 1984–1989. IEEE, 2006. (Cited on pages 16 and 20.)

[70] Adrian E Raftery. A model for high-order Markov chains. *Journal of the Royal Statistical Society. Series B (Methodological)*, 47(3):528–539, 1985. (Cited on page 49.)

[71] K N Ramachandran, E M Belding, K C Almeroth, and M M Buddhikot. Interference-aware channel assignment in multi-radio wireless mesh networks. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–12. IEEE, 2006. (Cited on pages 16 and 18.)

[72] A Raniwala, K Gopalan, and T Chiueh. Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *ACM SIGMOBILE Mobile Computing ...*, 2004. (Cited on pages 16 and 18.)

[73] Ashish Raniwala and Tzi-cker Chiueh. Architecture and algorithms for an IEEE 802.1 1 -based multi-channel wireless mesh network. In *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, volume 3, pages 2223–2234. IEEE, 2005. (Cited on pages 16 and 19.)

[74] Jochen Schiller. *Mobile Communications*. Addison-Wesley, 2nd edition, 2003. (Cited on page 8.)

[75] R Seibel, N H Klann, T Waage, and D Hogrefe. Wireless Mesh Networks for Infrastructure Deficient Areas. *Communications: Wireless in Developing Countries and Networks of the Future*, pages 26–38, 2010. (Cited on page 11.)

[76] C.E. Shannon. Communication in the Presence of Noise. *Proceedings of the I.R.E.*, 37(1):10–21, 1949. (Cited on page 8.)

[77] Weisheng Si, Selvadurai Selvakennedy, and Albert Y. Zomaya. An overview of Channel Assignment methods for multi-radio multi-channel wireless mesh networks. *Journal of Parallel and Distributed Computing*, 70(5):505–524, May 2010. (Cited on pages 2 and 17.)

[78] H Skalli, S Ghosh, S K Das, L Lenzini, and M Conti. Channel Assignment Strategies for Multiradio Wireless Mesh Networks :. *Communications Magazine, IEEE*, 45(11):86–93, 2007. (Cited on pages 16 and 17.)

[79] IEEE802.11 Std. IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Std. P802.11*, 1997. (Cited on page 1.)

[80] IEEE80211.a Std. Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications High-speed Physical Layer in the 5 GHz Band. Technical Report June, IEEE, 2003. (Cited on pages 66 and 67.)

[81] A P Subramanian, M M Buddhikot, and S Miller. Interference aware routing in multi-radio wireless mesh networks. In *Wireless Mesh Networks, 2006. WiMesh 2006. 2nd IEEE Workshop on*, pages 55–63. Ieee, 2006. (Cited on pages 26 and 28.)

[82] AP Subramanian, H Gupta, and SR Das. Minimum Interference Channel Assignment in Multi-Radio Wireless Mesh Networks. *Mobile Computing, IEEE Transactions on*, 7(12):1459–1273, 2008. (Cited on page 16.)

[83] M.K. Sue. Radio frequency interference at the geostationary orbit. Technical report, Jet Propulsion Lab., California Inst. of Tech., Pasadena., 1981. (Cited on page 8.)

[84] J Tang, G Xue, and W Zhang. Interference-aware topology control and QoS routing in multi-channel wireless mesh networks. *Proceedings of the 6th ACM international ...*, 2005. (Cited on pages 12 and 16.)

[85] SCALABLE Network Technologies. Qualnet, 2012. (Cited on page 20.)

[86] A. Tzamaloukas and J.J. Garcia-Luna-Aceves. A receiver-initiated collision-avoidance protocol for multi-channel networks. In *Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No.01CH37213)*, volume 1, pages 189–198. IEEE, 2001. (Cited on page 14.)

[87] Andrew Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on*, 13(2):260–269, 1967. (Cited on page 49.)

[88] IEEE802.11s WG. IEEE P802.11s/D8.0, Amendment 1: Mesh Networking. Draft Standard December, IEEE, 2010. (Cited on pages 2, 6, 25, 42, 65 and 72.)

[89] Kai Xing, Xiuzhen Cheng, Liran Ma, and Qilian Liang. Superimposed code based channel assignment in multi-radio multi-channel wireless mesh networks. In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking - MobiCom '07*, page 15, New York, New York, USA, September 2007. ACM Press. (Cited on pages 16 and 20.)

[90] Y Yang, J Wang, and R Kravets. *Designing Routing Metrics for Mesh Networks.* IEEE Comput. Soc. Press, 2005. (Cited on pages 11 and 26.)

[91] Yaling Yang, Jun Wang, and Robin Kravets. Interference-aware Load Balancing for Multihop Wireless Networks. *Technical Report, University of Illinois at Urbana-Champaign*, 361702:1–16, 2005. (Cited on pages 26 and 27.)

[92] Antonio Perez Yuste. Francisco Salva's Electric Telegraph. Technical report, Universidad Politécnica de Madrid, MAdrid, Spain, 2010. (Cited on page 6.)

[93] Weiyi Zhang, Farah Kandah, Jian Tang, and Kendall Nygard. Interference-Aware Robust Topology Design in Multi-Channel Wireless Mesh Networks. In *2010 7th IEEE Consumer Communications and Networking Conference*, pages 1–5. IEEE, January 2010. (Cited on pages 16 and 17.)

List of Figures

List of Tables

IaFI Intra Flow Interference . . . 2
IALM Intra-flow interference aware Airtime Link Metric . . . 47
iAWARE interference AWARE routing metric . . . 26
IBR Interferer-link Bit Rate . . . 28
IEEE Institute of Electrical and Electronics Engineers . . . 6
IP Internet Protocol . . . 6
IrFI Inter Flow Interference . . . 2
IR Interference Ratio . . . 28
IRU Interference aware Resource Usage . . . 27
ISI Inter-Symbol-Interference . . . 9
JOCAC Joint Optimal Channel Assignment and Congestion Control . . . 20
Kbps Kilo Bit per Second . . . 67
Kb Kilo Byte . . . 66
LA-CA Load Aware Channel Assignment . . . 18
LIBRA Load and Interference Balanced Routing Algorithm . . . 27
LOS Line-Of-Sight . . . 5
MAC Medium Access Control . . . 43
Mbps Mega Bit per Second . . . 10
Mhz Mega Hertz . . . 7
MIC Metric of Interference and Channel switching . . . 26
MIMC Multi-Interface Multi-Channel . . . 2
MR-LQSR Multi Radio Link Quality Source Routing . . . 21
MUP Multi-Radio Unification Protocol . . . 20
NP-hard Non-deterministic Polynomial-time hard . . . 16
NS-2 Network Simulator 2 . . . 18
NS-3 Network Simulator 3 . . . 43
OFDM Orthogonal Frequency-Division Multiplexing . . . 66
OLSR Optimized Link State Routing Protocol . . . 24
OSI Open Systems Interconnection . . . 6
PCU-CA Probabilistic Channel Usage based Channel Assignment . . . 19
PDF Packet Delivery Fraction . . . 71

PE	Peer Oriented	17
PERR	Path Error Packet	72
PSK	Phase Shift Keying	8
PMP	Peer Management Protocol	39
PREP	Path Reply Packet	21
PREQ	Path Request Packet	21
QAM	Quadrature Amplitude Modulation	66
QPSK	Quadrature Phase-Shift Keying	67
RDV	Route record based Distance Vector	3
R_C	Communication Range	8
RIP	Routing Information Protocol	24
R_I	Interference Range	3
RLR	Routing Load Ratio	71
RTT	Round Trip Time	25
SC-CA	Superimposed Code based Channel Assignment	20
SIM	Self Interference aware Metric	26
SIMC	Single Interface Multi Channel	14
SINR	Signal to Interference plus Noise Ratio	8
SISC	Single Interface Single Channel	12
SNR	Signal-to-Noise Ratio	8
SINR	Signal-to-Interference plus Noise Ratio	8
SRTT	Smoothed Round Trip Time	20
SS-CA	Self-Stabilizing Channel Assignment	20
TCP	Transmission Control Protocol	6
UDP	User Datagram Protocol	6
VCA	Varying Channel Assignment	16
WCETT	Weighted Cumulative Expected Transmission Time	21
WMN	Wireless Mesh Network	6

Personal information

Surname(s) / First name(s)	**Roman Seibel**
Address(es)	Sültebecksbreite.2, 37075 Göttingen, Germany
Telephone(s)	+49 176 20699758
Email(s)	roman.seibel@gmail.com
Nationality(-ies)	German
Date of birth	08/11/1981, Burnooktjabrskoje, Kasachstan
Gender	Male

Education

April, 2009 - October 2013

PhD., Computer Science
University of Göttingen, Germany
PhD Thesis: Route Record Distance Vector for Minimization of Intra-Flow Interference
Supervisors: Prof. Dr. Dieter Hogrefe, Prof. Dr. Xiaoming Fu

October, 2002 - March 2008

Diploma, Business Informatics
University of Göttingen, Germany
Diploma Thesis: Conception of a Telematics Demonstrator for Server based Point of Interest Download.
Supervisors: Prof. Dr. Dieter Hogrefe, University of Göttingen.
Mentor: Dr. Thomas Kleine-Besten, Bosch GmbH, Hildesheim.

Professional Experience

March, 2009 - December 2013

Research Fellow, University of Göttingen, Germany
Research: Wireless Multi-Channel Networks
Teaching: Lectures and Exercise Courses
Thesis Supervision: Master Thesis
Project acquisition support

May, 2008 - February 2009

Analyst, Accenture GmbH, Frankfurt am Main, Germany
Project: Development CRAMER Operating Support System
Client: Major Telecommunication Company

September, 2007 - March 2008

Work Student, Bosch GmbH, Hildesheim, Germany
Project: Prototype Development of Bluetooth Components for Intergrated Navigations Systems

Tools used: Cambridge Silicon Radio Embedded System and Python

August, 2006 - October 2006

Work Student, MahrUK Plc., Milton-Keynes, United Kingdom
Project: IT Infrastructure Assessment and Administration

November, 2005 - March 2006

Intern, Volkswagen Financial Services AG, Braunschweig, Germany

Project: SAP-Software development for Insurance Processes

August, 2005 - October 2005 **Work Student, DaimlerChrysler Inc, Rastatt, Germany**
Project: Division Controlling of Assembly Processes and Software Development

January, 2001 - March 2007 **System Administartor, Hainberg-Gymnasium, Göttingen, Germany**
Projects: Procurement, Implementation, Consultation

Teaching Experience

March, 2009 - August 2013 **Exercise Course - Foundations of Mobile Communications**
Topics: Fundamentals of Wireless Communication, Multiple Access, Telecommunication Systems, Security

November, 2009 - February 2010 **Lecture and Exercise Course - Computer Networks**
Topics: Transmission Control Protocol

January, 2010 - May 2010 **Supervised Master Thesis: Implementation of a Tool for Determining the Nominal Capacity of Wireless Mesh Networks**
Tim Waage, Master Thesis, No ZFI-MSC-2010-14, Center of Computational Sciences, University of Göttingen, Germany, ISSN 1612-6793

List of Publications

December 2012 **Autonomous Ant-based Public Key Authentication Mechanism for Mobile Ad-hoc Networks.**
Parisa Memarmoshrefi, Roman Seibel, Dieter Hogrefe
To appear in International Journal of Autonomous and Adaptive Communications Systems (IJAACS), 2012.

December 2010 **Bio-inspired Self-organized Public Key Authentication Mechanism for Mobile Ad-hoc Networks**
Parisa Memarmoshrefi, Roman Seibel, Dieter Hogrefe
In 5th International ICST Conference, BIONETICS 2010, Boston, USA, December 1-3, 2010, ISBN 978-3-642-32615-8

September 2010 **Wireless Mesh Networks for Infrastructure Deficient Areas**
Roman Seibel, Nils Klann, Tim Waage, Dieter Hogrefe
In The 3rd IFIP International Conference on Wireless Communications and Information Technology in Developing Countries (WCITD 2010), Brisbane, Australia, ISSN 1868-4238.

To be Published

2013 **Route Record Distance Vector Protocol for Minimization of Intra-Flow Interference in Multi-Channel Multi-Hop Networks**
Roman Seibel, Somayeh Taheri, Keyu Wang, Dieter Hogrefe
In preparation for publication in 2013