A SECURE AND RELIABLE INTERFERENCE-AWARE WIRELESS MESH NETWORK DESIGN

A Dissertation
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Farah Issa Kandah

In Partial Fulfillment of the Requirements
for the Degree of
DOCTOR OF PHILOSOPHY

Major Department:
Computer Science

December 2011

Fargo, North Dakota

ABSTRACT

A wireless mesh network (WMN) is a multihop wireless network consisting of a large number of wireless nodes of which some are called gateway nodes and connected with a wired network. Wireless mesh network have attracted much research attention recently due to its flexibility, low-cost and robustness, which facilitate its usability in many potential applications, including last-mile broadband Internet access, neighborhood gaming, Video-on-Demand (VoD), distributed file backup, video surveillance, *etc*. The broadcast nature, the lack of infrastructure as well as the flexible deployment nature of wireless mesh networks make it different from wired networks, therefore more attention in designing the wireless mesh network is needed to maintain a good performance of this promising technology. We, in this study, investigate the wireless mesh network design taking into consideration three design factors seeking an improvement in the network performance by reducing the interference influence in the network, improving the network reliability to satisfy more requests, and securing the network against malicious eavesdropping attacks. Our design is presented into three sub-problems; sub-problem (1), which seeks an interference-aware robust topology control scheme, sub-problem (2) which seeks a multipath routing scheme, and sub-problem (3) which seeks a secure key management scheme. Through simulations and comparisons with previous work, we show that our proposed solutions outperform previous schemes in providing a better network performance in terms of reducing the network interference, satisfying more number of requests and increasing the network resistance to malicious eavesdropping attacks.

ACKNOWLEDGMENTS

Foremost, I thank my family; my parents, my sisters and my lovely wife for supporting me spiritually through my life and my Ph.D. journey, "I owe them alot and will always be."

Beside my family, I would like to gratefully and sincerely thank my advisors Dr. Kendall Nygard and Dr. Weiyi Zhang for the continuous support during my Ph.D. study and research. Their guidance helped me all the time in my research and the writing of this dissertation. I could not have imagined having better advisors and mentors for my Ph.D study. Also I would like to thank my colleague Mr. Yashaswi Singh for his help through this research.

Finally, many thanks to the rest of my dissertation committee members; Dr. Jun Kong, Dr. Tariq King and Dr. Chao You, for their encouragement, their feedback, their expertise and insightful comments.

DEDICATION

To my father and my mother.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

CHAPTER 1. INTRODUCTION

A wireless mesh network [2, 8, 14, 50, 51, 61] is a multihop wireless network consists of a large number of wireless devices, such as mesh routers, mesh gateways and mesh end users. Wireless mesh networks have attracted much research attention recently, due to their flexible easy deployment, low cost and robustness. These properties and others facilitate the usability of these networks in many potential applications, including last-mile broadband Internet access, neighborhood gaming, Video-on-Demand (VoD), distributed file backup, video surveillance [8, 61, 65].

In this study, we investigate and propose a wireless mesh network design taking into consideration three design factors; (interference influence, reliability, and security), thus seeking an improvement in the performance of wireless mesh networks. Previous studies have shown that maintaining small interference in the network can provide a well and robust wireless systems [50, 51, 59, 58, 61]. In fact, with a well designed channel assignment among wireless nodes, a better network performance can be provided in terms of capacity and network throughput [7, 58]. One common technique used to improve overall network capacity in multi-hop wireless networks is the use of multiple network interface cards with multiple channels. The idea behind this is instead of using a single channel in multi-hop wireless networks, multiple channels can be used to improve the network throughput dramatically as well as increasing the network capacity which is brought mainly by allowing multiple simultaneous transmissions within a neighborhood [50, 51]. The IEEE 802.11a and IEEE 802.11b standards offer 3 and 12 non-overlapping channels respectively. In a single channel wireless network, two transmissions in a neighborhood are not allowed to happen at the same time because of the contention for the shared wireless channel. However, in a multi-channel network, no collision will be caused by such simultaneous transmissions as long as they work on different channels.

With to the large number of users and the emergence of real-time multimedia applications, providing reliability have become critical issues in WMNs. We, in this study consider the network reliability as our second design factor. Previous studies showed that multipath routing was able to improve the network reliability in WMNs [62]. Multipath routing aims to find several disjoint paths for a specific connection [48], i.e., instead of finding one path for a connection, several disjoint paths can be found to reach the destination. Therefore, when any link or node failure happens on the primary path, all the information can still be transmitted using other paths. To satisfy users' requests and improve the network reliability, each request is accommodated by two disjoint paths, a primary and a protection paths. The protection path is reserved (not actively used) for a request until a failure occur at the primary path. The idea of multipath routing is shown in Figure. 1. The red solid lines show the primary path, which is active all the time, and the blue dashed lines show the protection path. If a failure occur in the primary path, the transmissions will be transferred over to the protection path without interruption or notifying the users of any failure in the path.

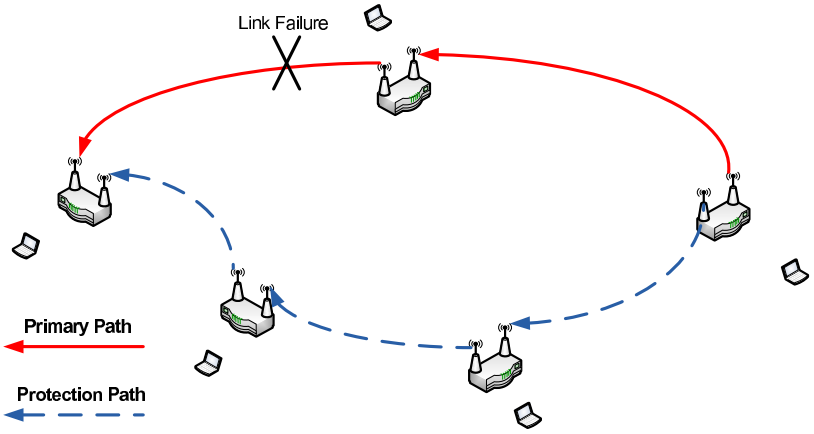


Figure 1: Link-disjoint primary and protection paths (an example)

Due to the flexible deployment nature and the lack of fixed infrastructure, WMNs had suffer from a variety of security attacks [23, 72]. The existence of such attacks might hold back the potential advantages and wide scale deployment of this promising wireless

network technology. Furthermore, the broadcast nature of wireless networks [63] compared to wired networks increases their vulnerability to adversary attacks. WMNs could suffer from several kind of attacks, such attacks could be passive, active or a combination of both passive and active attacks. After compromising a node by an adversary, it will be easy for it to access all the node's contents including the encryption keys and use them for its own good. In the case of passive attacks [69], the adversary after compromising a node and accessing all its information, will start eavesdropping on all the messages in its transmission range without letting any other node notice its existence. To improve the network design we, in this work, focus on the network security as our third design factor taking into consideration the malicious eavesdropping attacks as our main attack to stand against.

1.1. Organization of the Dissertation

The rest of this dissertation is organized into five chapters. The literature review is provided in Chapter 2 including background, related work and our motivations behind this study. Followed the literature review chapter are three chapters other chapters. Each chapter focuses on a specific design factor, presents a solution, and analyzes the effectiveness and the efficiency of the proposed solution. Considering our first design factor, Chapter 3 presents the Interference-aware Robust Topology scheme. In this chapter we formally state our sub-problem (1) and present our solution and show using numerical results its effectiveness and efficiency. Chapter 4 covers our second design factor. In this chapter, we present our diverse path routing scheme and show through numerical results the improvement in the network performance by applying our multipath routing scheme. We, in Chapter 5, present the secure key management problem as our third design factor and discuss our proposed solution and show through simulation the effectiveness and the efficiency of this proposed solution. Finally we conclude this study in Chapter 6.

CHAPTER 2. LITERATURE REVIEW

2.1. Background

2.1.1. Wireless Networks

Wireless networks refers to any type of networks that consist of a set of wireless nodes that are not connected by cables of any kind. These networks are used in telecommunications networks and enterprize (business) to avoid the costly process of introducing cables into a building, or as a connection between various equipped locations [22]. Wireless communications among different wireless nodes are generally implemented and administered using radio waves, which takes place at the physical level (layer) of the network structure [22].

Wireless networks are being increasingly used due to its ability to create communication among devices of various types and sizes. Personal computers, personal digital assistants (PDA), telephones, appliances, industrial machines, sensors, and others are being used in several environments, such as residences, buildings, cities, forests, and battlefields. Recent years have seen different wireless network standards and technologies to enable easy deployment of applications in such networks [44, 52].

The lack of infrastructure rises some challenges in wireless networks, where ad-hoc networks have been proposed to solve such problems. A **wireless ad-hoc network** consist of a number of wireless nodes with the ability of dynamically self-organizing into an arbitrary and temporary topology to form a network without the necessity of using any pre-existing infrastructure. The broadcast nature of wireless networks allow wireless transmissions, where each node in wireless ad-hoc network can communicate directly to any node within its transmission range. On the other hand, nodes that are not within the transmission range of each other can still communicate through multi-hop, where nodes in such networks, have the ability to work as routers. The main advantages of ad-hoc networks are flexibility, low cost, and robustness. Minimal configuration and quick deployment

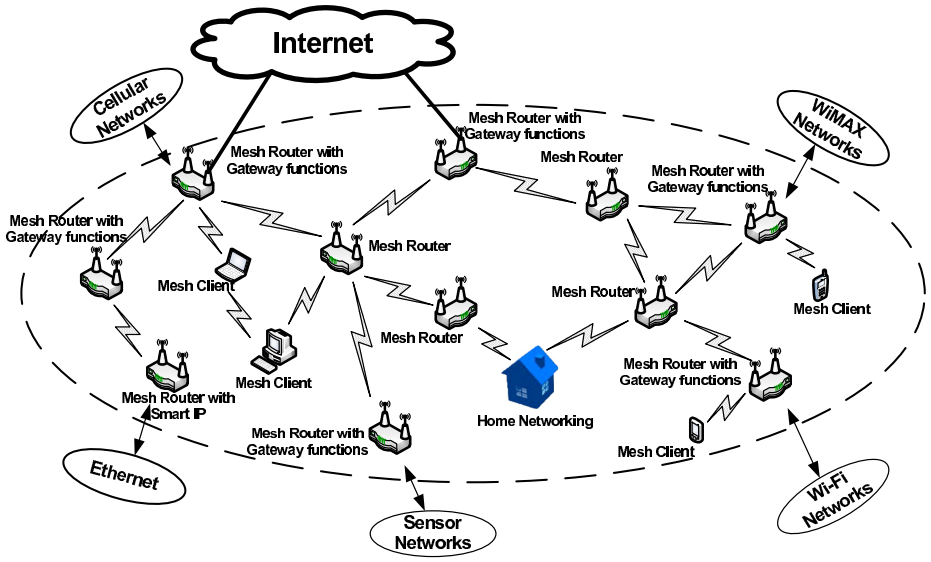


Figure 2: Wireless mesh network (an example)

make ad-hoc networks suitable for emergency situations like natural disasters and military conflicts. In fact the presence of dynamic and adaptive routing protocols enable ad hoc networks to be formed quickly.

Wireless mesh networks (WMNs) are dynamically self-organized and self-configured networks, where nodes in the network are able to automatically establish an ad-hoc network and maintain the mesh connectivity. A wireless mesh network [2, 50, 51] is a multi-hop wireless network consists of a large number of wireless nodes, some of which are called gateway nodes and connected with a wired network. Mesh nodes are small radio transmitters that are able to perform as wireless routers, with the use of the common Wireless Fidelity (WiFi) standards known as 802.11a, 802.11b and 802.11g to communicate wirelessly with users, and with each other [30]. WMNs has attracted much research attention recently due to its potential applications, including last-mile broadband Internet access, neighborhood gaming, Video-on-Demand (VoD), distributed file backup, video surveillance and so on [2]. An example of WMN is presented in Fig. 2.

Most traditional wireless access points require a wired connection to the Internet to broadcast their signal. In large wireless networks, Ethernet cables need to be buried in walls, ceilings and throughout public areas. On the other hand, wireless mesh network require only one node to be physically wired to a network connection (Internet). That one wired node is able to broadcast wirelessly the Internet connection with all other nodes in its vicinity, and those nodes will share the connection wirelessly with the nodes closest to them. The more nodes, the larger the area that can be covered with connection, which creates a wireless "cloud of connectivity" that can serve a small office or a city of millions.

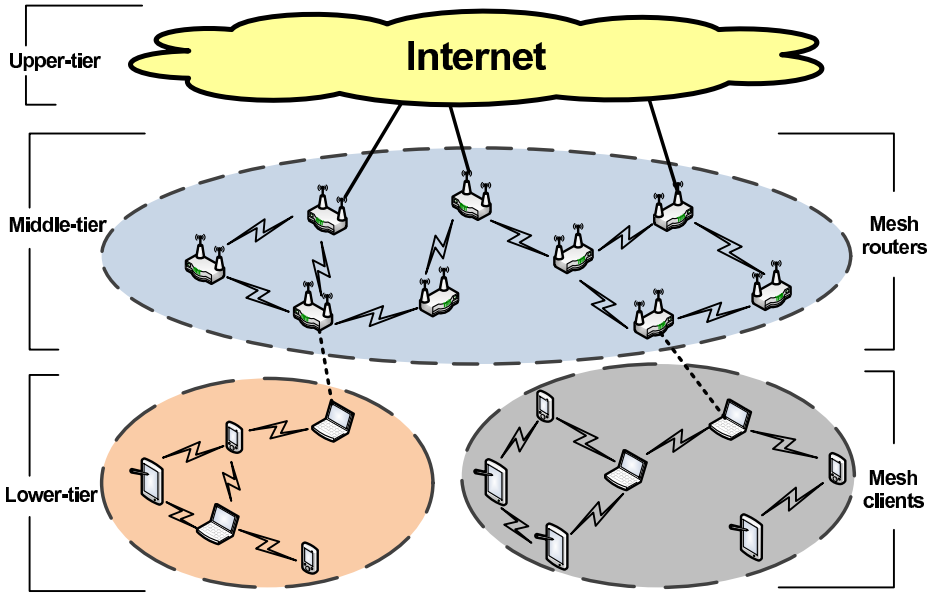


Figure 3: Wireless mesh network architecture

The architecture of WMNs can be classified into three groups based on the mesh nodes' functionality. An illustrated example of this classification is presented in Fig. 3.

- **Infrastructure/Backbone WMNs:** Mesh routers form an infrastructure for clients as well as a mesh of self-configuring, self-healing links among themselves. This is represented at the middle-tier of Fig. 3. Mesh routers are able to form a cloud of connectivity between them, thus spread the connection among the network. Mesh

routers (referred to as gateway mesh routers) has the ability to be connected directly to the interned using wired links.

- **Client WMNs:** In this architecture, the actual network is formed of client nodes, where they performed routing and configuration functionalities as well as providing end-user application to customers. This architecture is portrayed in Fig. 3 at the lower-tier. This architecture is referred to as pure mesh [45].

- **Hybrid WMNs:** This is a combination of both Infrastructure and client WMNs architectures. Clients with wireless network interface cards (WNIC) can connect directly to WMNs through wireless mesh router. While clients without WNIC can connect to the wireless mesh routers through Ethernet [2]. Thus, WMNs allow users to be connected to the Internet anywhere, anytime. With this architecture wireless mesh clients can access the network through wireless mesh routers as well as other mesh clients.

The above architectures of WMNs provide an intelligent organization of a high bandwidth ad-hoc network which is driven by the user and application needs. We summarize the key characteristics of WMNs in the following:

- **Self-configuring, self-managing and self-healing:** each node works out the routing itself to form the network automatically. In the case of a node failure, other nodes will remove the routes shared the failed node and establish new routes to automatically manage the network. The existence of obstacles such as walls and buildings could block the wireless signal. Wireless mesh nodes are able to avoid any fading in the wireless signal by self-adjusting to find a clear signal.

- **Dynamic changes in network topology:** self-configuring provides these networks with the ability to manage any change in the mesh topology when nodes are added,

removed, replaced or relocated. In other words, the network automatically integrate a new node into the existing structure without the need of any adjustments by a network administrator.

- **Scalability:** Since the routing configuration in mesh networks is automatic, the size of the network can be increased by adding more number of nodes with no exponential rise in complexity.

Therefore, WMNs diversify the capabilities of ad-hoc networks instead of simply being another type of ad-hoc network. These additional capabilities required new algorithms and design principles for the realization of WMNs.

2.1.2. Radio Link

Wireless transmissions are carried over Radio Frequency (RF) links which provide the transmission medium for wireless devices. The coverage area and capacity depend on the attributes of the RF links. Previous studies have shown that there are practical distance limits for which a signal can be reliably communicated, which can be calculated as how much strong a signal is at the receiver. Practical studies showed that the signal strength decreases with distance due to noise and interference [7]. In fact, a well and robustness wireless system design strongly depends on the ability to maintain an adequate signal to noise ratio and interference over the entire coverage area.

Wireless nodes have built in transmission antennas designed to transform a signal into an electromagnetic waves and propagate them into the surrounding environment. They also have receiving antennas to capture the electromagnetic waves and transform them back to signals. An electromagnetic wave of a specific frequency provides the channel between transmitting and receiving antennas at wireless nodes.

Co-channel interference (CCI) occur when two different radio transmitters are transmitting at the same time using the same channel frequency. Co-channel radio interference can occur in wireless networks for different reasons. For example, in cellular mobile

communication, frequency spectrum is divided into non-overlapping spectrum bands. Geographical areas are divided into cells, where each cell refers to the hexagonal/circular area around the base station antenna. Each cell will be provided with non-overlapping spectrum bands. However, the frequency bands are re-used after a certain geographical distance. As a result of frequency reused, the co-channel interference arises in cellular mobile networks. I.e., signals at the same frequencies (co-channel signals) arrive at the receiver from the undesired transmitters located (far away) in some other cells and lead to deterioration in receiver performance.

2.1.3. Reliability and Survivability

Wireless and mobile services have attracted many communication and realtime multimedia applications recently, but the ability of wireless network infrastructures to handle the growing demand is questionable. Wireless networks are more prone to failure compared to wired networks [36]. Failures in such networks could affect current voice and data use and limit emerging wireless applications such as e-commerce and high-bandwidth Internet access [37, 39]. The networks ability to avoid or cope with failure in wireless (and wireline) networks could be measured by any of the following.

- **Reliability:** which is defined as the ability of the network under certain conditions to perform a designated set of functions for specified operational times.
- **Availability:** which is the networks ability to operate and be in committable state at any given instant under certain conditions. Average availability is a function of how often something fails and how long it takes to recover from a failure.
- **Survivability:** which is defined as the capability of a system to fulfill its mission, in a timely manner, in the presence of attacks, failures, or accidents.

It is worth nothing that by taking into consideration any of those measurements, the network performance could be improved significantly.

2.1.4. Encryption Key Management

With more attentions on wireless ad-hoc networks lately, the security issues become more important and urgent for managing and deploying in such networks [72]. The flexible deployment nature and the lack of fixed infrastructure make these networks suffer from a variety of security attacks [23, 72], where the existence of such attacks might hold back the potential advantages and wide scale deployment of this promising technology. The adversary could have the ability to compromise an arbitrary number of nodes, through physical capture or software bugs, thus gaining full control of them. Once compromised, the adversary will extract all the security information stored in the compromised nodes as well as the encryption keys preloaded into their memories. On the other hand, if the adversary was not able to physically compromise a node and due to the broadcast nature of wireless networks, the adversary could have the ability to capture any message that is being sent by any node in its neighborhood. Most of the current security mechanisms (e.g., encryption and digital signature) applied to WMNs are based on cryptographic keys and thus providing a well designed key management services are in demand [4]. Key management service responsibilities include establishing a trusted secure communication between nodes as well as keep track of bindings between keys [23]. It is worth nothing that the way in which keys are assigned among nodes in the network could make the network resistant or vulnerable to malicious attacks. Therefor, a well designed key management scheme could increase the network resistance against malicious attacks.

2.2. Related Work

Recent researches have shown that the interference influence in wireless networks can make a significant impact on the network performance. As a pioneering work, Gupta and Kumar in [26] showed that in a wireless network with n identical nodes, the per-node throughput is $\Theta(1/\sqrt{n \log n})$ by assuming random node placement and communication pattern. It becomes $\Theta(1/\sqrt{n})$ under the assumption of optimal node placement and com-

munication pattern. In [31], the authors modeled the influence of interference using a conflict graph and derive upper and lower bounds on the optimal throughput. Burkhart et al. gave a concise and intuitive definition of interference in [9], which shows that most proposed topology control algorithms do not effectively constrain interference. The problem of channel assignment in such networks can be seen as a graph multi-coloring problem [41, 47]. Narayanan in [47] studied the channel assignment using the graph multi-coloring, where he modeled a cellular data and communication networks using graphs with each node representing a base station in a network cell and edges representing geographically adjacency of cells. Co-channel interference were modeled in terms of reuse distance taking into consideration minimizing the total number of channels used over all node in the network. The authors in [41] proposed a weighted colored based channel assignment technique to improve the usage of wireless spectrum in the context of wireless networks. Using 3 non-overlapping channels, the authors showed that their proposed technique provided an interference reduction of 50% less compared to previous work.

Recently, people begin to study multi-channel multihop wireless networks, such as multi-channel wireless mesh networks and multi-channel Mobile Ad-hoc NETwork (MANET), since network throughput can be substantially improved by making full use of the non-overlapping channels. One of the first IEEE 802.11-based multi-channel multihop wireless mesh network architectures was proposed and evaluated in [51]. The authors developed a set of centralized algorithms for channel assignment, bandwidth allocation, and routing. They also presented distributed algorithms utilizing only local traffic load information to dynamically assign channels and to route packets in a later paper [50]. Draves et al. in [18] presented a new metric named Expected Transmission Time/Weighted Cumulative (ETT/WCETT), for multiradio, multihop wireless networks which can be used for finding a high-throughput path between a source and a destination. Ding *et al.* in [17] proposed a channel assignment scheme to improve the network throughput. Their proposed

scheme (the Greedy algorithm) consists of a series of decisions to assign a channel to a link. Their decision composed of two steps; the select step and the assign step. Each link will be assigned a specific value (α), where a link will be chosen among all the links correspondingly. After selecting the link, all the channels will be assigned another value (β), and according to this value, a channel will be assigned to the selected link. Another study in [61] proposed a robust network topology design with the consideration of interference. The authors in [35] proposed algorithms for channel assignment and routing in multi-channel multi-NIC mobile ad-hoc networks (MANETs). Vaidya et al. [56] also presented a routing protocol for the scenario where each node has only one NIC. Besides routing protocols, several link layer and MAC layer solutions have been proposed for multi-channel multihop wireless networks in [5]. We note that previously proposed channel assignment schemes are not suitable for dynamic traffic model since no traffic demand profile is available as the guideline for channel assignment. Moreover, none has considered network robustness and survivability.

The ability of multipath routing schemes in providing a better quality of service to transfer multimedia applications such as voice, video and data, has been proved in a number of previous work [11, 46, 67]. Chen *et. al* in [11], addressed the problem of real-time video streaming over a bandwidth and energy constrained in wireless sensor network. Due to these constraints, the authors proposed to divide a single video stream into multiple sub-streams, and use multiple disjoint paths to transmit these sub-streams in parallel. The authors presented a directional geographical routing (DGR) scheme allowing the use of these parallel sub-stream in an efficient way that facilitates load balance and provides bandwidth aggregation as well as fast packet delivery. Through simulations they showed that their proposed scheme provided longer network life time and better received video quality. Wu *et. al* in [67] presented a multipath routing scheme (Ad hoc on-demand multipath routing) seeking a better quality of service in terms of bandwidth, hop count

and end-to-end delay in mobile ad-hoc networks. In their proposed scheme, the authors provided an alternative path that will be used as a next primary path to continue data transmission without initiating a route discovery in the case if a failure in the main primary path due to node mobility. Through simulations they showed that their multipath routing scheme provided high reliability and low overhead in the network.

In [46] the authors showed that multi-path routing design can improve the reliability of packets delivery by providing many alternate loop-free paths to destination. Mohanoor *et al.* in [43] studied a way to improve the end-to-end throughput in wireless networks by the use of diverse paths with less interference. To improve route recovery and control the message overhead in wireless sensor networks for indoor environments, the authors in [29] proposed a routing scheme using multiple node-disjoint paths. Tsai and Moors in [62] studied a multi-path routing design and focused on the concurrent use of multiple paths. In their scheme, they sent copies of data over different paths to improve the end-to-end reliability. The authors presented a multipath selection heuristic algorithm that will exploit the frequency diversity offered in a multi-radio, multi-channel network. In [53] the authors studied the use of multipath routing by using concurrent paths between two nodes to increase the effective throughput. In [57], the authors studied the problem of finding minimum energy disjoint paths in wireless ad-hoc network. They presented a heuristic algorithm where for each request, after finding a primary path, they used all the nodes along the primary path (the common nodes) to find another path that shared these common nodes to form a link-disjoint path. Hu and Lee in [28] proposed a multipath routing protocol named AODV-based decoupled multipath routing protocol (AODV-DM), aiming to find multiple paths with less interference. After finding a primary path, an insulating region is formed around the primary path, which contains all the edges within the interference range of each node on the primary path. A protection path must be selected and established outside the insulating region to reduce potential network interference with the primary path

found. We realized that most of the previous work assumed to divide the network load on two disjoint path to improve the quality of service. However, any existence of interference between links within the disjoint path might degrade the network performance as well as degrades its capacity. Moreover, previous work indicated that a protection path will be used in the case of a failure in the corresponding primary path. It is worth nothing that the protection links can be reused to protect multiple primary paths if some criteria are satisfied.

Recent researches have shown that security attacks are holding back the potential advantages and wide-scale deployment of wireless networking technology [23, 72]. Several key management schemes [19, 23, 38, 72] have been proposed for wireless networks and claimed to have high security. However, their weakness such as high computational overhead, storage overhead and vulnerability to some kinds of attacks are undeniable. Du *et al.* in [19] proposed a key management scheme for heterogenous sensor networks. Each high-end sensor is preloaded with M keys, and each low-end sensor is preloaded with L keys ($M >> L$) in a pre-distribution phase, where the keys are randomly picked from a pool of keys (P) without replacement. This phase will be followed by the discovery phase to check the availability of common keys between any pair of neighboring sensors, and the key setup phase, which is used to provide a shared key to any sensors pari with no common keys shared between them. Note that, this proposed scheme could be affected by the pool size. With a large pool size and a small K keys randomly selected from (P) to be stored in each node, a better security can be provided [19]. Moreover, with small pool size, there will be a chance of having more nodes shared common keys in a same neighborhood which might harm the network in the case of an adversary existence. We observed that in small network neither all the generated keys in the pool nor the keys in high-end or low-end sensors are being used or needed.

Zhao *et al.* in [72] proposed an elliptic curve cryptosystem (ECC)-based self-certified public key cryptosystem. Their scheme provided an efficient authentication and key agreement between mesh clients and routers. Another key management scheme was proposed in [21], where the authors designed a key management scheme with selective distribution and revocation of keys to sensor nodes as well as node rekeying. In their proposed scheme a key pre-distribution phase required a large pool (P) of keys (*e.g.,* 2^{17} - 2^{20} keys), where each sensor's memory is preloaded with a random drawing of K keys out of (P) without replacement. Shared key discovery phase follows the previous phase to establish the topology, where every node discovers its neighbor in its wireless communication range with which it shared keys. A low-computational and scalable key management model for WMNs was proposed in [23], where the authors aim to guarantee a well performed key management service and protection against potential attacks. Another study in [10] considered the problem of designing a key management scheme in a clustered distributed sensor networks, where the probability of node compromise in different deployment regions is known in advance. In their scheme, the network with n nodes is divided into S subgroups, in which, each node within a subgroup is preloaded with a set of keys using the scheme in [21]. Different probability of node compromise values are assigned to different subgroup.

2.3. Motivations

Recent studies have shown that interference can make a significant impact on the performance of wireless networks. Multi-channel multihop wireless networks have attracted much research attention recently due to their higher performance compared to single channel wireless networks. To achieve a better network performance, multiple channel assignment schemes were proposed seeking to reduce the interference influence in such networks. The motivations behind our first design factor can be summarized in the following.

- The authors in [9, 42] proposed an adjustment in the transmission power at each node to minimize the impact of interference in wireless networks. However, this proposed scheme required a change in the default settings of the wireless nodes. To avoid this shortcoming, we in this work assume that the transmission power at each node is fixed, and no change on IEEE 802.11 distributed coordinated function (DCF) is required [31].

- Previous studies in [5, 26, 51, 56] provided a channel assignment schemes based on available traffic demand profile as a guideline for channel assignment. We realized that the lack of previous knowledge of the network traffic make it hard to apply previous schemes. In this work, our proposed channel assignment scheme is suitable for dynamic traffic, where no previous knowledge existed about the traffic in the network. We believe the dynamic traffic model is more useful in reality because considering the aforementioned applications in the future, we should expect not only some traffic from wireless nodes to the Internet via gateway nodes but also substantial random and unpredictable traffic among wireless nodes within the mesh network, where it might be hard to precisely predict the traffic demands in advance.

- Dynamic and hybrid channel assignment schemes were proposed in [13, 15, 27, 66]. The authors in dynamic schemes assigned channels for NICs in the route discovery phase and allow channel switching during packet transmissions. This is not very suitable for wireless mesh networks, because dynamic assignment may cause the deafness problem (transmitter and intended receiver happen to be on different channels) and need fine grained synchronization [35]. Moreover, with the hybrid channel assignment schemes, the authors assigned one NIC of each node statically to a common control channel, and allowed other interfaces to dynamically switch among other data channels. We realized that this might avoid the deafness problem but might hold back the throughput improvement, especially in the case where the

number of NICs in each node is very small. Note that, the channel switching delay of current commercial IEEE 802.11 hardware is in the range of a few milliseconds to a few hundred microseconds [35], which is intolerable for most real-time multimedia applications. We, in this work consider a static channel assignment among all the nodes in the network to avoid previous designs' shortcomings.

Due to large number of users and the emergence of multimedia application, reliability and survivability become mandatory issues to be provided in wireless networks. Multipath routing schemes have shown their ability to improve wireless networks performance in transferring multimedia applications such as, voice, video and data [12, 29, 43, 46]. The motivations behind our second design factor can be summarized as follows.

- Due to the limited number of frequency channels, the interference influence in wireless networks can be reduced but might be hard to eliminate [60]. Wu *et. al* in [67] proposed to provide a an alternative path that will be used as a protection path to continue data transmission without initiating a route discovery. All the transmissions will be carried on the primary path which is active all the time. In the case of failure in the primary path, the transmissions will be moved to the protection path without loosing the connection between the source and the destination. The authors in [28] realized that network interference can affect the multipath performance, where they proposed a multipath routing protocol with less interference by constructing an insulating region around the primary path where the protection path will be chosen outside this insulating region. It is worth nothing that by constructing an insulating region around the primary path, a large number of network links might be eliminated form the possibility of being a part of other paths which in turn might reduce the number of available multipaths in the network. We observed that by embracing the network interference a better protection performance can be provided, i.e., since the primary paths are active all the time, therefor they should not use the interfered

links because the interference could affect the bandwidth on those primary paths. Therefore, we proposed to choose the interfered links with the primary path to be part of the protection paths.

- We realized that in multipath routing each protection path is reserved (not actively used) for a request in the case of a failure in the primary path. Therefore, it is possible to use a same link to be a part of multiple protection paths if some criteria are satisfied.

Recent researches have shown that security attacks are holding back the potential advantages and wide-scale deployment of wireless networking technology [23, 72]. Several key management schemes [19, 23, 38, 72] have been proposed for wireless networks and claimed to have high security. However, their weakness such as high computational overhead, storage overhead and vulnerability to some kinds of attacks are undeniable. The motivations behind out third design factor can be summarized in the following.

- In practice, when a node is compromised by an adversary, all the information stored in that node will be extracted by that adversary including the set of encryption keys preloaded to that node. We realized that, the way the keys are assigned to/among all the nodes in the network could make the network resistant or vulnerable to malicious attacks. Due to the broadcast nature of wireless networks, when any neighboring node of node u, say v, send an encrypted message to any other node outside the transmission range of node u, say w, then this message can be decrypted by node u if node v encrypted the message using any encryption key $k \in keys(u,v) \bigcap keys(v,w)$, where $keys(u,v)$ is the set of keys shared between node u and node v and $keys(v,w)$ is the set of keys shared between node v and node w. We observed that previous key management schemes did not consider the effect of sharing the same keys between nodes within a 2-hops neighboring range of a any node. Therefor, we proposed

a secure key management scheme ensuring the secure connectivity and taking into consideration the 2-hop neighboring range effect when assigning encryption keys to the nodes in the network.

- Previous work [19, 21] indicated that to assign keys to nodes in a network, a large pool (P) of keys must exist, from which a set of keys ($K|K \in P$) is chosen randomly to be assigned to each node. We realized that neither all the generated keys in the pool nor all the keys stored in some nodes are needed or being used. Moreover, the ratio between the pool size and the number of keys $|K|$ could affect the network. I.e., with large pool size, more variety of keys will be available to chose from. If the process of generating encryption keys for the pool of keys is expensive, this will affect the key assignment scheme, where the variety in choosing keys from the pool will be small, and that in turn will lead to having the same keys being shared between multiple nodes in the same neighborhood. To avoid previous shortcomings, we provided a secure key management scheme using $|K|$ keys to be assigned among nodes in the network, without the need to generate large pool of keys.

CHAPTER 3. INTERFERENCE-AWARE ROBUST TOPOLOGY CONTROL

IEEE 802.11 [30] Wireless Local Area Networks (WLANs) offer low cost, ease of installation, and fast setup features. These networks provide relatively high-performance wireless accesses, and are widely adopted at homes, in offices, and hot-spot areas in cities. Due to the limited channel capacity, the influence of interference, the large number of users and the emergence of real-time multimedia applications, improving network capacity have become a critical requirement in such networks. One common technique used to improve overall network capacity is the use of multiple channels. Previous studies have shown an improvement in the network throughput in wireless networks using multiple channels [50, 51], where instead of using a single channel in multihop wireless networks, multiple channels will be used to allow simultaneous transmissions in the same neighborhood. The IEEE 802.11b standard and IEEE 802.11a standard offer 3 and 12 non-overlapping channels respectively. We, in this work, considered a multi-channel wireless mesh network, where every node is equipped with multiple Network Interface Cards (NICs) and each of them is assigned to a distinct frequency channel [49, 58, 61].

Two neighboring mesh nodes can communicate with each other if they have NICs using the same channel. The architecture of a multi-channel wireless mesh network is shown in Fig. 4. The network capacity improvement in such networks is brought mainly by allowing multiple simultaneous transmissions within a neighborhood. In a single channel wireless network, two transmissions in a neighborhood are not allowed to happen at the

The material in this chapter was co-authored by Farah Kandah (North Dakota State University), Weiyi Zhang (AT&T - Research Labs), Jian Tang (Syracuse University) and Kendall Nygard (North Dakota State University). Farah Kandah had primary responsibility for the algorithm design, the implementation an the extraction of the simulation results. The original work was presented and published at IEEE Consumer Communications & Networking Conference (CCNC) 2010, Las Vegas, NV [70]. Farah Kandah also drafted and revised all versions of this chapter. Weiyi Zhang, Jian Tang and Kendall Nygard served as proofreaders and checked the logic and the math in the design conducted by Farah Kandah.

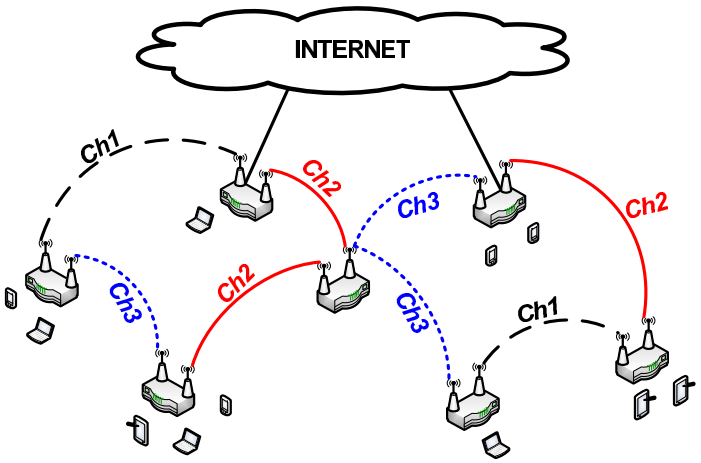


Figure 4: Multi-channels wireless mesh network architecture (an example)

same time because of the contention for the shared wireless channel. However, in a multi-channel network, no collision will be caused by such simultaneous transmissions as long as they work on different channels.

How to assign channel for each NIC is a basic issue in such networks. We adopt a static assignment scheme [50, 51, 61], i.e., assigning channels for each NICs before connection requests arrive and keep the computed assignment for a long period of time. Once a channel assignment is given, the network topology can be determined. Intuitively speaking, we want the channels assigned to the NICs in a common neighborhood to be as different as possible such that interference can be reduced. In addition, we need to preserve the network connectivity and support survivability. In this paper, based on the novel definition of co-channel interference which can capture the impact of interference precisely [61], by fully considering both interference and connectivity, we define the **I**nterference-**A**ware **R**obust **T**opology (**I-ART**) problem which seeks a network topology design and a channel assignment such that the induced network topology has the minimum network interference among all 2-connected topologies. In this work, 2-connectivity is required for survivability and load-balancing purposes. We assume the transmission power of each NIC is fixed. So the topology control problem studied here is quite different from all

previous topology control problems [9, 42] in which the network topology is controlled by carefully adjusting the transmission power at each node. Like the other higher layer solutions proposed in [35, 50, 51], our scheme can be used without making any change on IEEE 802.11 DCF.

The rest of this chapter is organized as follows. We describe the system model and formally define the problem in Section 3.1. Our interference-aware robust topology control scheme is presented in Section 3.2 followed by the numerical results in Section 3.3.

3.1. Problem Statement

In this section, we first describe our system model and notations. Then, we formally define the optimization problem we are studying. Note that, the terms edges and links are used interchangeably hereafter. We use a similar network model as described in [50, 51, 61]. There are totally $\mathcal{C}$ non-overlapping frequency channels in the system and each node is equipped with Q NICs where $Q \leq \mathcal{C}$. In order to efficiently and fully make use of the network resources, we assume that each NIC is tuned to a channel and that any two NICs at the same node are tuned to different channels. All nodes in the network use the same fixed transmission power, i.e, there is a fixed transmission range ($r > 0$) and a fixed interference range $R > r$ (which is typically 2 to 3 times of r [50] associated with every node). We use an undirected graph $G(V, E)$ to model the wireless mesh network where V is the set of n vertices and E is the set of m edges. Each vertex $v \in V$ corresponds to a stationary wireless node in the network with its location known. There is an undirected edge (u, v) $\in E$ connecting vertex u and vertex v if $d(u, v) \leq r$, where $d(u, v)$ is the Euclidean distance between u and v. The edge (u, v) in G corresponds to a potential wireless link between nodes u and v in the network. Note that wireless link (u, v) cannot be realized until node u and v are assigned with the same channel.

Definition 1. Potential interference edge: Given any two edges (u, v) and (x, y) in G, if node x or y is in the interference range of u or v (covered by the disk centered at u or v

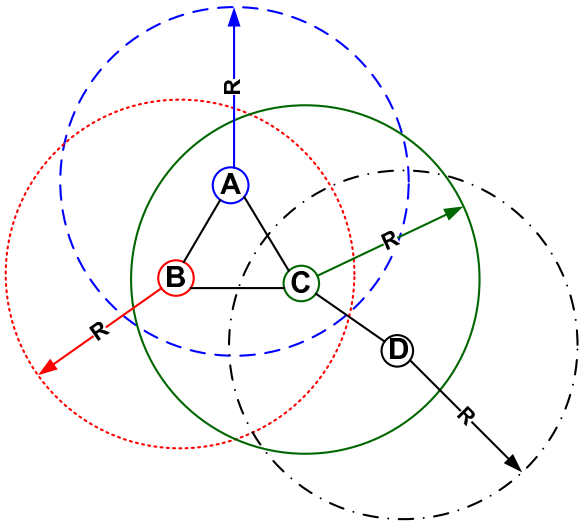


Figure 5: Potential interference edge

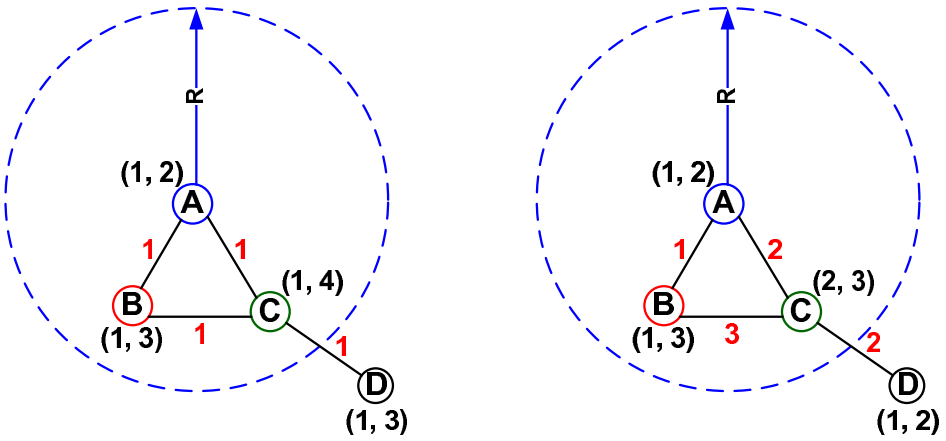
with radius R), then we say that edge (x, y) is a potential interference edge of edge (u, v), and vice versa.

We use Fig. 5 to illustrate the definition of the potential interference edge. Nodes A, B and C are 1-hop neighbors to each other, while node D is node C's 1-hop neighbor. Each node has an interference range with radius (R). The interference range of each node is depicted using large circles with corresponding colors. Since node C is inside the interference range of node A and B, node C's transmissions have potential interference with nodes A and B's transmissions. On the other hand, any transmission from node A or node B has a potential interference with node C's transmissions. By considering this situation, we can say that edge (C, D) is a potential interference edge of edge (A, B), and vice verse.

A channel assignment $\mathcal{A}$ assigns each node $v \in V$ a set $\mathcal{A}(v)$ of Q different channels: $A(v) \subseteq 1, 2, \ldots, \mathcal{C}$. The channels in $\mathcal{A}(v)$ correspond to the $\mathcal{C}$ different channels that the Q NICs at node v are tuned to. A channel assignment $\mathcal{A}$ defines a topology $G_{\mathcal{A}}(V, E_{\mathcal{A}})$ in the following natural way: There is an edge $e = (u, v; k)$ on channel $\lambda(e) = k$ between nodes u and v in $G_{\mathcal{A}}$ if and only if $d(u, v) \leq r$ and $\lambda(e) \in \mathcal{A}(u) \cap \mathcal{A}(v)$.

Definition 2. Interference edge: Given a channel assignment $\mathcal{A}$ and its corresponding network topology $G_{\mathcal{A}}$, for any two potential interference edges (u, v) and (x, y), if there is a channel $k \in \mathcal{A}(u) \cap \mathcal{A}(v) \cap \mathcal{A}(x) \cap \mathcal{A}(y)$, then the link $(u, v; k)$ interferes with the link $(x, y; k)$, since simultaneous transmissions along $(u, v; k)$ and $(x, y; k)$ will lead to collision. $(x, y; k)$ is an interference edge to $(u, v; k)$, and vice versa.

It is worth noting that before channel assignment is given, it is impossible to calculate the interference suffered by each edge. However, because the interference edges will be a subset of the potential interference edges, if we can reduce the number of potential interference edges, we could reduce the interference in the network. Let e be a link in G, we will use *PIE*(e) to denote the set of potential interference edges of e and *PIN*(e) to represent the edge's potential interference number, the size of the set *PIE*(e). Similarly, *IE*(e) and *IN*(e) are used to denote the set of interference edges of e in $G_{\mathcal{A}}$ and interference number of e, respectively.



(a) Simple channel assignment (b) A better channel assignment

Figure 6: Channel assignment (an example)

We will use Fig. 6 to illustrate the definition of interference edges. Note that, we extend our example in Fig. 5 by assigning channels to the potential interference edges. In Fig. 6, each node is equipped with two NIC with two different channels assigned to

its NICs. To achieve connectivity, each node will communicate with its neighbor using a common channel from its assigned channel. In Fig. 6(a), link (A, B) interferes with link (C, D) because node C is inside A's interference range and both links are using channel 1. Actually all the links in Fig. 6(a) are interfering with each other since all the links are potential interference edges to each other and using the same channel. A better channel assignment in terms of less interference is shown in Fig. 6(b). It can be seen that only edges (A, C) and (C, D) interference with each other in this case. Note that, the more diverse the channels are in a neighborhood, the more simultaneous transmission are allowed with less interference.

Definition 3. Network Interference: Given a network G, and a channel assignment $\mathcal{A}$, the network interference is defined as the maximum edge interference number among all the edges in network $G_{\mathcal{A}}$, which is $\max_{e \in G_{\mathcal{A}}} IN(e)$.

Table 1: Channel assignment (1)

Edge	***PIE(e)***	***IE(e)***	***IN(e)***
(A, B)	(A, C), (B, C), (C, D)	(A, C), (B, C), (C, D)	3
(A, C)	(A, B), (B, C), (C, D)	(A, B), (B, C), (C, D)	3
(B, C)	(A, B), (A, C), (C, D)	(A, B), (A, C), (C, D)	3
(C, D)	(A, B), (A, C), (B, C)	(A, B), (A, C), (B, C)	3

From our example in Fig. 6, we can calculate the network interference for each channel assignment as shown in Tables 1 and 2. The potential interference edges (*PIE*), the interference edges and the interference number for each edge in the network, given the channel assignment in Fig. 6(a) are presented in Table. 1. For example, considering edge (A, B), all edges in this example are considered as potential interference edges for edge (A, B). With the channel assignment given in Fig. 6(a), edges (A, C), (B, C) and (C, D) are considered as interference edges with edge (A, B) since they are assigned the same channel. From the information given in Table. 1, the network interference is calculated

as the maximum interference number among all edges in the network, therefore in this example **the network interference is 3**.

Table 2: Channel assignment (2)

Edge	***PIE(e)***	***IE(e)***	***IN(e)***
(A, B)	(A, C), (B, C), (C, D)	-	0
(A, C)	(A, B), (B, C), (C, D)	(C, D)	1
(B, C)	(A, B), (A, C), (C, D)	-	0
(C, D)	(A, B), (A, C), (B, C)	(A, C)	1

The corresponding results of the channel assignment given in Fig. 6(b) are presented in Table. 2. It can be seen that by assigning the channels in a neighborhood to be as different as possible while keeping the network connected, we can reduce the interference influence by having less number of links that share the same channels in the same neighborhood. Note that different channel assignments can induce different corresponding topologies, and different network interferences. Given the channel assignment in Fig. 6(b) **we can reduce the network interference to 1** compared to the one given by the channel assignment in Fig. 6(a).

We formalize our **I**nterference-**A**ware **R**obust **T**opology (**I-ART**)control problem in the following.

Definition 4. I-ART Problem: Given the network G, the Interference-Aware Robust Topology (**I-ART**) problem seeks a channel assignment $\mathcal{A}$ such that the corresponding network topology $G_{\mathcal{A}}$ is 2-connected (robust to any single failure) and has the minimum network interference.

It is worth noting that by reducing the network interference, I-ART aims to minimize the maximum interference suffered by any edge in the network to improve the network performance. We try to assign the channel evenly throughout the network, and consequently generate a more robust and balanced network.

3.2. Interference-aware Robust Topology Control Scheme

I-ART is complicated because channel assignment and network topology can affect each other. Without network topology, it is impossible to assign channels on the edges and calculate network interference. On the other hand, different channel assignments can induce different network topologies. Our idea is to split the I-ART problem into two sub-problems, by finding a robust survivable network topology and an interference-aware channel assignment. Correspondingly, our solution in carried out in two steps. **First** we try to find a 2-connected network topology with minimum number of edges. At this time, all the edges are virtual edges because no channel has been assigned. Only potential interference edges exist in the network at this step. **Secondly**, we try to assign channels on each edge to realize the network topology, with the goal to minimize the interference influence in the network.

Our solution is listed in Algorithm 1. In the first step (Line 1), we aim to find a subgraph G' of network G, where G' is expected to be 2-connected and has the minimum number of edges. The idea behind this is that the less number of edges in the network, the less potential interferences between edges. Consequently, the less network interference. It is not hard to see that a Hamiltonian cycle will be the ideal minimum 2-connected network. But it is well known that finding a Hamiltonian cycle in a network is a NP-hard problem [24]. Therefore, we use an effective linear time algorithm in [71] to find a 2-connected structure (a pair of redundant trees) in the given network G. Researchers from Massachusetts Institute of Technology (MIT) presented an elegant scheme known as redundant trees scheme [40] to construct a pair of directed spanning trees from a common root node in a way that the failure of any edge (or a node other than the root node) in the network leaves each node still connected with the root node using at least one of the trees, provided that the network is 2-connected. They named one of the trees the red tree and the

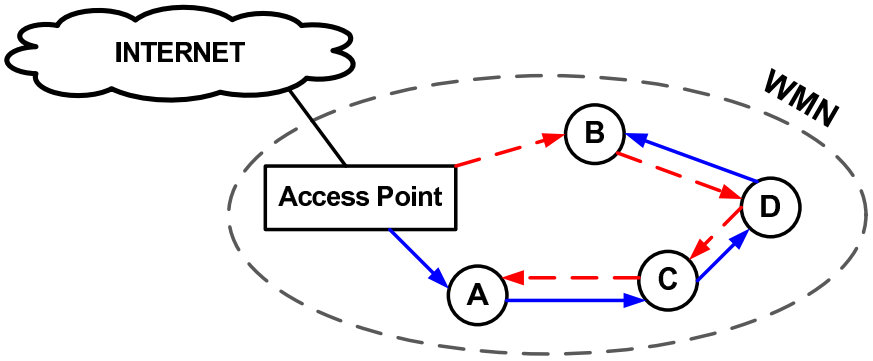


Figure 7: Red/Blue redundant trees

other the blue tree. They showed that, for any 2-connected network, there exists a pair of red/blue trees which can provide fast recovery from any single link or node failure.

Let us use Fig. 7 to illustrate the concept of redundant trees. Starting with the root node, the access point to the Internet, a pair of directed trees is constructed. The red tree is depicted by the red dotted lines, while the blue thick solid lines form the blue tree. Assume that a failure occur at a node, say A, consequently, nodes B, C and D cannot be reached by the root on the blue tree. However, all the remaining nodes in the network are still connected with the root node through the red tree.

In Line 1 of Algorithm 1, a pair of redundant trees form a 2-connected subgraph of the network, which can survive any wireless node failure in the network. Another advantage of redundant trees is that the route from any node to the access point (the root of the trees) is also decided after the trees are constructed. Linear-time algorithm **ReducedCostV** [71] is adopted to construct a pair of red/blue trees with the near-minimum number of edges in Line 1, which means that the near-minimum number of potential interference edges in the network. Next step we aim to find a channel assignment to realize the network topology with minimum interference. Ideally, all the potential interference edges will be assigned with different channels. Between Line 2-3, we first calculate the potential interference edges (*PIE*) for each edge of G, and also have the *PIN* value of the edge. All the NICs on

Algorithm 1: Interference Aware Robust Topology Control

input : $G(V, E)$
output: Channel assignment

1 Find 2-connected subgraph of G, $G'(V, E')$ such that G' has the *minimum number of edges*;
2 **for** *each link* e *in* G' **do**
3 Find the *PIE*(e) and calculate the *PIN(e)* of e;
4 Initialize $\mathcal{A}(u)$ to $\emptyset$ for all $u \in V$;
5 **for** *all the links in* G' **do**
6 Select links one by one *in a descending order of PINs*;
7 For the selected link $e \in G'$, assign channels for all edges in *PIE*(e) based on the following rules:
8 **for** *all end nodes of edges in PIE(e)* **do**
9 Select the nodes in *PIE*(e) one by one *in a descending order of node degree*;
10 **if** *there are* $l(\geq 1)$ *empty NICs on the selected node* u **then**
11 Use the l least used channels to fill all the empty NICs on node u;
12 **for** *all* unassigned edges $e' = (u, v)$ *where* v *has empty NICs* **do**
13 Assign the currently least used one among the l channels to edge e';
14 Assign corresponding channel on node v;
15 **for** *all* unassigned edges (u, v) *where* v *has* NO *empty NIC* **do**
16 Channel Swap $(G, u, v, (u, v))$;
17 **if** *(No empty NIC on node* u*)* **then**
18 **for** *all* unassigned edges $e' = (u, v)$ *where* v *has empty NICs* **do**
19 Assign the currently least used channels on u on e';
20 Assign corresponding channel on node v;
21 **for** *all* unassigned edges (u, v) *where* v *has* NO *empty NIC* **do**
22 Channel Swap $(G, u, v, (u, v))$;

all the nodes are initialized to be empty (Line 4). Then we assign channels on the edges and nodes cluster by cluster. Here each cluster is a set of edges including an edge and its potential interference edges. We pick the edges, consequently the clusters, according to the descending order of edges' *PIN* values. The edge with the highest *PIN* is the first edge to be processed. Then we try to assign channels for all the edges in the cluster. After selecting a cluster, for all the nodes in the cluster, we start channel assignment from the node with

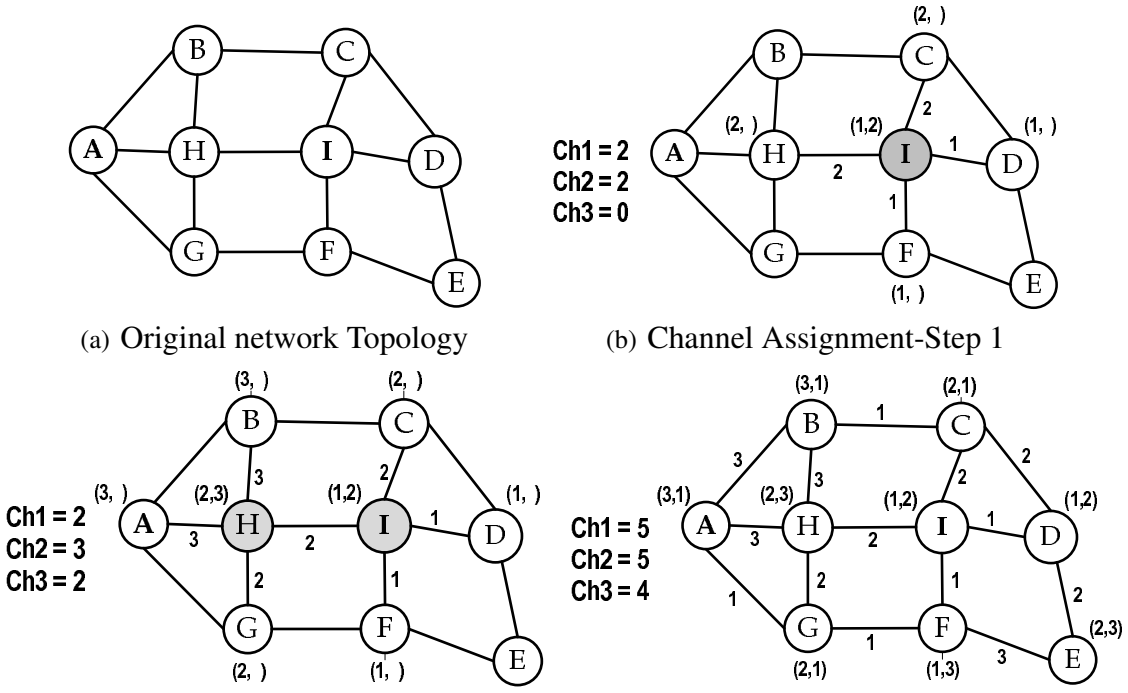
highest node degree (Line 9), say node u. If there are l empty NICs on u, we will use the l least used channels in the cluster, and try to distribute them evenly over the neighboring nodes, and consequently the edges. One special case which should be considered is that when a neighboring node v does not have any empty NIC. Then, the channel selected by the node u cannot be used on node v. If this case happens, we use Algorithm 2 to find the best common channels for u, v and edge (u, v).

Algorithm 2: Channel Swap

input : $G(V, E)$
output: Channel swap
1 **if** $\mathcal{A}(u) \cap \mathcal{A}(v) \neq \emptyset$ **then**
2 continue;
3 **else**
4 let k be the least used channel in *PIE*(*e*) among channels in $\mathcal{A}(u) \cup \mathcal{A}(v)$. Without loss of generality, assume that $k \in \mathcal{A}(u)$.
5 Let $k' \neq k$ be a channel in $\mathcal{A}(v)$ that is most used in *PIE*(*e*).
6 Replace k' in $\mathcal{A}(v)$ by k.
7 **for** *all the edges* (v, w) *already assigned* **do**
8 **if** *the change of* $\mathcal{A}(v)$ *makes* $\mathcal{A}(v) \cap \mathcal{A}(w) = \emptyset$ **then**
9 Replace k' in $\mathcal{A}(w)$ by k.

In Algorithm 2, if nodes u and v already share some common channel, we can use this channel on the edge (u, v) (Line 1-2). Otherwise, we need to replace some channel either on u or on v to find a common channel to be used on (u, v). To achieve that, we find the least used channel among all the channels assigned on node u and node v. Then we use this channel to replace the most used channel on the other node (Lines 4-9 of Algorithm 2).

We use an example in Fig. 8 to illustrate our algorithms. For simplicity, we assume that all the links are in the same interference range in this example. Each node is equipped with 2 NIC and we will use 3 channels to be assignees among all the nodes in this example. The network topology is shown in Fig. 8(a). According to Algorithm 1, we start with the node that has the most number of unassigned edges. Node I and node H have the same



(a) Original network Topology (b) Channel Assignment-Step 1

(c) Channel Assignment-Step 2 (d) Channel Assignment-Final

Figure 8: Illustration of the interference-aware robust topology control scheme

number of unassigned edges, and neither of them has any channels on their NICs. By considering the nodes' first degree (1-hop neighbors) and second degree (2-hop neighbors) we will choose node I as a start point since it has a higher node degree. First we choose one of the unassigned edges (I, D), then we pick the least used channel from the available channels, say channel 1. We assign channel 1 on the edge and update the channel usage and add the corresponding channel on node D's NIC. We repeat the previous steps for the next unassigned edge (I, C) using the current least used channel, say channel 2, then we update the channel usage and add it to node C's NIC. At this time both NICs on node I are filled, therefore, to minimize the network interference, we evenly assigned channels 1 and 2 on edges (I, H) and (I, F), as shown in Fig. 8(b). After assigning all the edges of node I, we choose the next node from *PIE* with the highest node degree. Here we will continue our channel assignment with node H. We start assigning channels on all unassigned edges

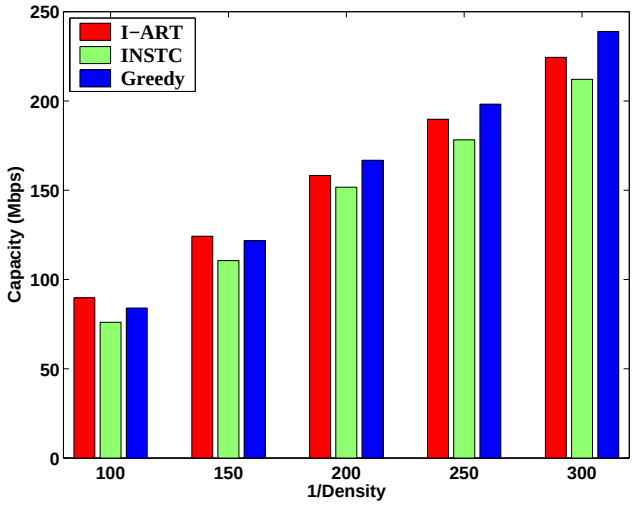
of node H, following Algorithm 1, Lines 10-14. The channel assignment at this step is shown in Fig. 8(c). By following the same steps befrore, we can assign channels on all the nodes and the edges as in Fig. 8(d). The worst case running time for Algorithms 1 and 2 is $O(nm^2)$, where n is the number of nodes and m is the number of edges.

3.3. Numerical Results

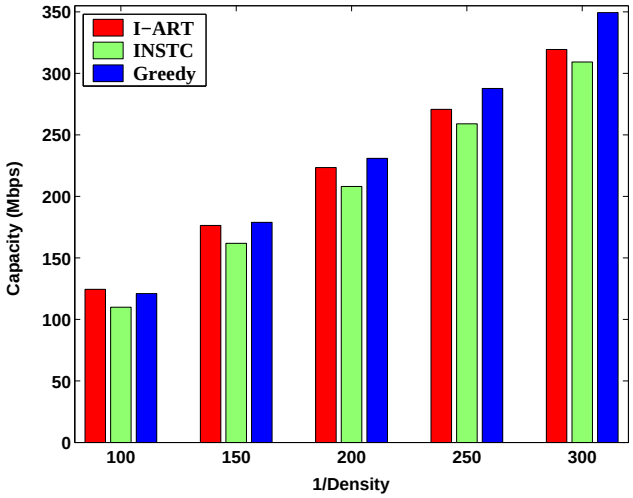
In this section, to illustrate the performance of our proposed scheme, we implemented our solution (denoted by **I-ART** in the figures), and compared it with the channel assignment schemes in [61] (denoted by INterference Survivable Topology Control (**INSTC**) in the figures) and in [17] (denoted by **Greedy** in the figures). Static wireless mesh networks with n nodes uniformly distributed in a square playing field were considered as in [61]. In all simulation scenarios, we require 2-connectivity to be preserved. According to IEEE 802.11 specifications, we set the number of channels to be 3 (802.11b) and the corresponding channel capacity to 11*Mbps*. The number of network interface card (NIC) at each node is set to 2. Each node has a fixed transmission range of 250 meters and an interference range of 500 meters [50]. Our simulations are realized using LEDA 4.2 [25]. The results shown are the average of 20 test runs for various scenarios.

The first matric used for performance evaluation is the **network capacity**. For each edge e, we calculate its bandwidth as the channel capacity (Cap) divided by the number of interfere edges of e, *IN*(e). In short, the bandwidth of e is $\frac{Cap}{IN(e)}$. Note that this is the best case estimation for the network in terms of bandwidth allocation fairness. The network capacity was calculated as the summation of the bandwidths of all edges in the network. We realized that the more number of nodes in a square unit, the more dense the network will be. Therefor, by increasing the number of nodes in one square unit, the number of edges in the network will increase as well, since nodes in a neighborhood will be close to each other. This increase in the number of edges, will increase the interference influence in the network. We in this work consider the network density as a base of our performance

evaluation, which is defined as the number of nodes in one square unit. In short, the density equals to $\frac{number\ of\ nodes}{Area\ size}$.



(a) 200 nodes



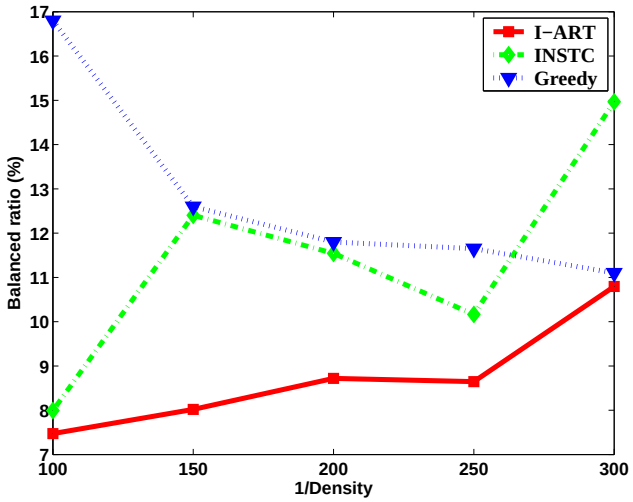
(b) 300 nodes

Figure 9: Network capacity versus density

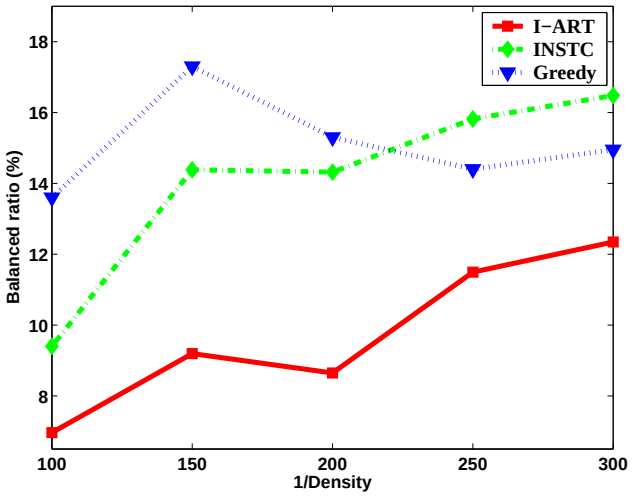
To show the performance of the network using our proposed channel assignment scheme (I-ART) and compare it to that using the channel assignment schemes in [17] and [61], we distribute 200 and 300 nodes in five different area sizes to evaluate the performance in dense and sparse networks. The corresponding results are presented in Fig.

9. Our first scenario's results in terms of network capacity are shown in Fig. 9(a) which illustrate the growth of the network capacity as the network size increases. It can be seen that our I-ART scheme produces more network capacity compared to the INSTC scheme. Based on our results, the network topology generated by I-ART has less number of edges than the network topology generated by INSTC, which reduces the chances of potential interferences and consequently produces more total network capacity. Also it can be seen that our I-ART scheme provides a higher network capacity in dense network compared to that with the Greedy scheme in [17]. For example, with 200 nodes in an area size of 2×10^4 square meters (*density* = $\frac{1}{100}$), our I-ART scheme provides a network capacity of 89.7 *Mbps* compared to that of 76.02 *Mbps* with the INSTC scheme and 83.9 *Mbps* with the Greedy scheme. The same results' trend can be seen in Fig. 9(b) where we distributed 300 nodes in different area sizes. We realized that increasing the network density by increasing the number of nodes will increase the network capacity due to the increase in the number of available edges in the network. It can be seen that with 300 nodes in 2×10^4 square meters area size, we can provide 124.5 *Mbps* by applying our I-ART scheme, compared to that of 121.07 *Mbps* with the Greedy scheme and 109.9 *Mbps* with the INSTC scheme.

To minimize the interference influence in the network we aim to assign channels among links in the network to be as even as possible in order to provide a balanced channel assignment to the network, which can provide us with a balanced bandwidth among the network, rather than having some spots in the network with high bandwidths and other spot with low bandwidth. To show how much balanced the network is, we define our second performance metric the **balanced ratio** which is calculated as the ratio of the maximum edge bandwidth among all the edges over the minimum edge bandwidth among all the edges. Note that, the smaller the ratio, the more balanced the channel assignment is, which reflects a reduction in the interference influence in the network. We show the balanced ratio for 200 nodes and 300 nodes distributed in five different area sizes in Fig. 10.



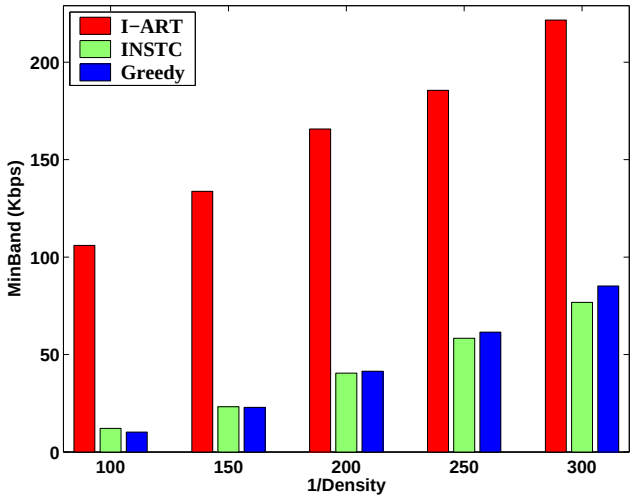
(a) 200 nodes



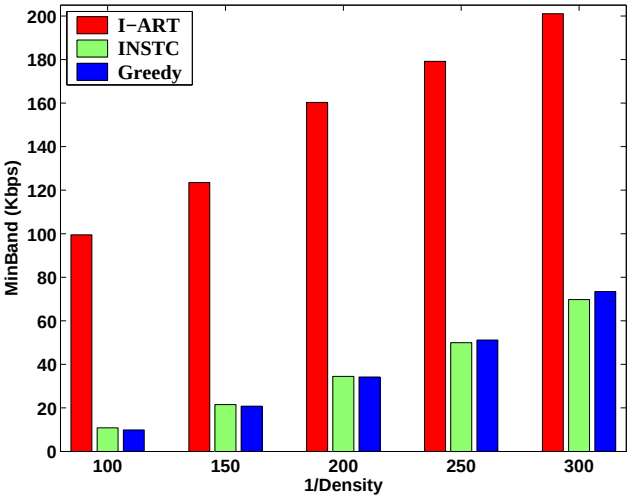
(b) 300 nodes

Figure 10: Balanced ratio versus density

It can be seen that our I-ART scheme outperforms the previous schemes in terms of providing a smaller balanced ratio, which implies that the channels were evenly assigned among the edges in the network. For example, in Fig. 10(a) with 200 nodes in 3×10^4 (*density* = $\frac{1}{150}$) square meters area size, our I-ART scheme assign channels to the network with a balanced ratio of 8.02%, while the INSTC scheme's provide a balanced ratio of 12.4% and the Greedy scheme's provides a balanced ratio of 12.6%.



(a) 200 nodes



(b) 300 nodes

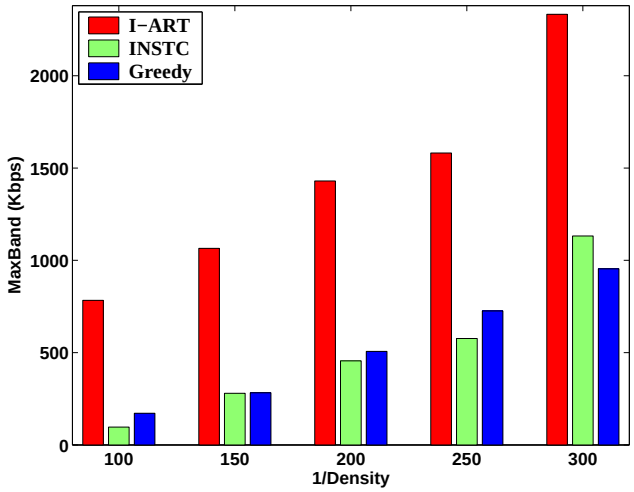
Figure 11: Minimum bandwidth versus density

We evaluate the performance of our scheme in dense network and compare it to the other two scheme in [17] and [61], by distributing 300 nodes in five different area sizes. Our results are presented in Fig. 10(b). It is obvious that our I-ART scheme performs well in dense network and provides a better balanced ratio compared to the other two schemes. For example, in 3×10^4 (*density* = $\frac{1}{150}$) square meters, the INSTC and the

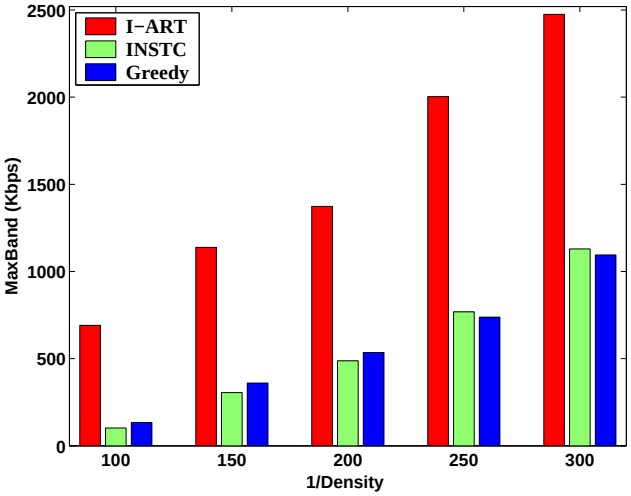
Greedy schemes provide a balanced ratio of 14.38% and 17.3% respectively, while our I-ART scheme assigns the available channels with a balanced ratio of 9.19%.

Our third performance metric is the **minimum bandwidth** (denoted as MinBand in the figures) which is defined as the smallest link's bandwidth among all the links in the network. After channel assignment, each link's bandwidth can be specified ($\frac{Cap}{IN(e)}$). Note that, in dense networks with smaller area sizes, nodes will be close to each other which will increase the number of links in the network. Therefore, with limited number of available channels and large number of links in a network, the chance of sharing the same channel on multiple edges will increase, and that will affect the network in having smaller links' bandwidth. We tested our I-ART scheme in two different scenarios, where 200 and 300 nodes were distributed in five different area sizes. The corresponding results in terms of minimum bandwidth are presented in Fig. 11. It can be seen from Fig. 11(a) that our I-ART scheme outperform the other two schemes in providing a higher bandwidth, due to the smaller number of edges in the network which result in minimizing the chance of sharing the same channel on large number of edges. For example, with 200 nodes in 2×10^4 square meters area size, our I-ART scheme provides a minimum bandwidth of 105.9 *Kbps* compared to that of 12.11 *Kbps* with the INSTC scheme and 10.2 *Kbps* with the Greedy scheme. Our results in Fig. 11(a) show that the minimum bandwidth increases in sparse network compared to that in dense network, due to less number of links sharing the same channels compared to that in dense networks. For example, in 4×10^4 square meters area size (*density* = $\frac{1}{200}$), our I-ART scheme provides a minimum bandwidth of 165.7 *Kbps* compared to that of 40.4 *Kbps* with the INSTC scheme and 41.45 *Kbps* with the Greedy scheme. The same results' trend can be seen in Fig. 11(b) where we distributed 300 nodes in the same area sizes as in Fig. 11(a).

Our fourth performance metric we consider in this chapter is the **maximum bandwidth** (denoted MaxBand in the figures) which is defined as the maximum available band-



(a) 200 nodes

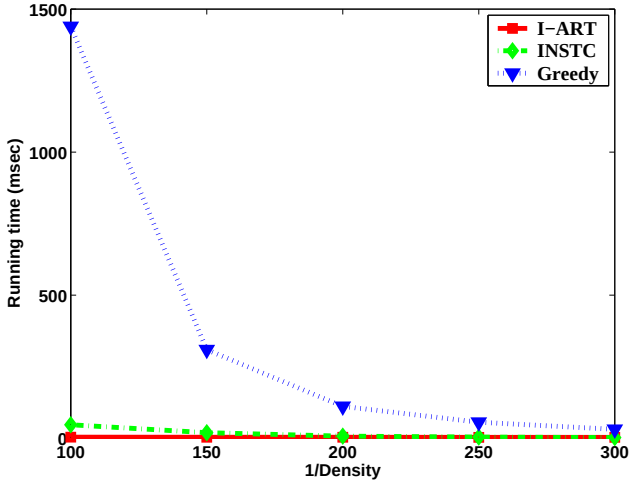


(b) 300 nodes

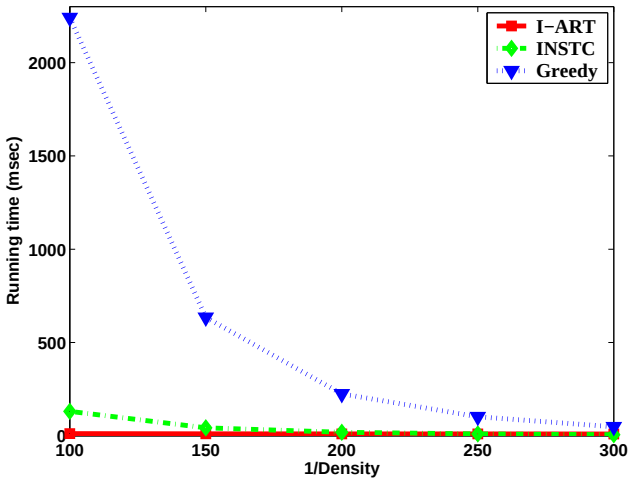
Figure 12: Maximum bandwidth versus density

width among all the links in the network. The corresponding results are shown in Fig. 12. It can be seen from Fig. 12(a) that our I-ART scheme outperform the other two schemes in providing a higher bandwidth compared to the other two schemes, due to the smaller number of edges in the network which result in minimizing the chance of sharing the same channel on large number of edges. For example, in Fig. 12(a), with 200 nodes in 3×10^4 square meters area size, our I-ART scheme provides a maximum bandwidth of 1064.97

Kbps compared to that of 279.9 *Kbps* with the INSTC scheme and 282.6 *Kbps* with the Greedy scheme.



(a) 200 nodes



(b) 300 nodes

Figure 13: Running time versus density

To support our results, we test the performance of our proposed scheme in dense networks and compare the results to that when applying the INSTC and the Greedy schemes. Fig. 12(b) present the results of our second scenario, where 300 nodes were distributed in five different area sizes. It is obvious that our I-ART scheme provides a higher bandwidth

compared to that with the other two schemes. For example, with 300 nodes in an area size of 4×10^4 square meters (*density* = $\frac{1}{200}$), our I-ART scheme provides 837.5 *Kbps* and 885.73 *Kbps* more bandwidth on the links, compared to that of the Greedy and the INSTC schemes, respectively.

Finally, we consider the **running time** as our last performance metric, which is defined as the time it takes for a scheme to assign channels among all the links in the network. We distribute 200 and 300 nodes in five different area sizes and present our results in Fig. 13. Our results show that our I-ART scheme consumes less time to assign channels among the links in the network compared to the other two scheme. Note that, before assigning a channel on a link, say e using the Greedy scheme, all the links in the neighborhood will be checked in order to choose the least used channel to be assigned on link e in order to minimize the interference influence. This process will consume too much time especially in dense network where the number of links is higher compared to that in sparse network. This can be seen in Fig. 10(a) where 200 nodes were distributed in five different area sizes. For example, in 2×10^4 square meters area size, our I-ART takes 3.77 *msec* to assign channels among all the links in the network, while the INSTC scheme takes 47.09 *msec* and the Greedy scheme takes 1438.97 *msec*. The same results' trend can be seen in Fig. 10(b) where 300 nodes where distributed in five different area sizes.

CHAPTER 4. DIVERSE PATH ROUTING

Due to the large number of users and the emergence of real-time multimedia applications, providing reliability has become one of the critical issues in WMNs. Using multiple paths instead of single path routing has been proven to improve the reliability as well as the quality of service in wireless mesh networks [62]. The idea of multipath routing is instead of finding one path for a connection, we find several disjoint paths to reach the destination [48]. Therefore, when any link or node failure occurred on the primary path, all the information could still be transmitted using other protection paths.

We in this chapter focus on our second design factor. We realized that the interference influence in the network can be reduced but it might be hard to eliminate. Therefor, to improve the network reliability, we aim to embrace the network interference for better protection performance. First, we observed that any two primary paths should not use the interfered links because the interference will reduce bandwidths of both primary paths. Our second observation is that a primary path and its protection path will never transmit at the same time. And so, instead of using a link in an insulating region [28], we plan to use the links which are interfered with the links on a primary path for protection. To the best of our knowledge, this is the first work to consider the use of network interference to improve the connection accommodation in wireless mesh networks.

In this chapter, we study routing in WMNs with dynamic traffic, i.e., users' requests have random arrival times, which is different from the static network routing which was studied in [50, 51, 61], where all traffic demands were given in advance. We strongly

The material in this chapter was co-authored by Farah Kandah (North Dakota State University), Weiyi Zhang (AT&T Research Labs), Chonggang Wang (InterDigital Communications, LLC), and Juan Li (North Dakota State University). Farah Kandah had primary responsibility for the algorithm design, the implementation an the extraction of the simulation results. The original work in this chapter was published in the Mobile Networks and Applications (2011) journal [34]. Farah Kandah also drafted and revised all versions of this chapter. Weiyi Zhang, Chonggang Wang, and Juan Li served as proofreaders and checked the logic and the math in the design conducted by Farah Kandah.

believe that this dynamic request model is more useful in reality because considering the users requests in the future, we should expect not only the previously known requests but also the request that could come to the network at the runtime. For each coming request we need to provide two link-disjoint paths (one primary path and one protection path) to satisfy the user's request, i.e., by providing each request with two link-disjoint primary and protection paths, we can say that we had satisfied the request [33]. Each primary path will be reserved for a specific user request. On the other hand, each protection path is reserved (not actively used) for a request in the case of a failure in the primary path. It is possible to use a same link to protect multiple primary paths if some criteria are satisfied. For example, if we assume single-link failure in a network, then a link can be used as a part in different protection paths to protect multiple active paths as long as they are link-disjoint. We denote such ability to protect multiple paths as reusability of a protection link. To the best of our knowledge, this work is the first to discuss the reusability of protection links in wireless mesh networks.

The rest of this chapter is organized as follows. We describe the system model and formally define the problem statement in Section 4.1. Our diverse path routing scheme is presented in Section 4.2, which is followed by the numerical results in Section 4.3.

4.1. Problem Statement

First in this section, we will describe our system model and notations. Then, formally will define the optimization problem we are going to study. Note that, the terms edge and link are used interchangeably, as well as the terms diverse path and *multipath*. As our network model, we used a similar network model as described in [50, 51, 61]. All nodes in any given network will use the same transmission range (r), where $r > 0$ and an interference range (R), which is typically 2 to 3 times of the transmission range (r) that associated with each node [50]. We used an undirected bi-connected graph $G(V, E)$ to model the wireless mesh network where V is the set of n nodes and E is the set of m

links in the network. For each pair of nodes (u, v), there exist a undirected edge $e \in E$ if and only if $d(u, v) \leq r$, where $d(u, v)$ is the Euclidean distance between u and v. Each edge between any pair of nodes (u, v) in G corresponds to a potential wireless link between nodes u and v in the network.

Definition 5. Interference edge: Given any two edges (u, v) and (x, y) in G with a given channel assignment, if node x or node y is in the interference range of node u or node v (Lies within a distance R from node u or node v), and they have been assigned the same channel k, then we can say that edge (x, y) is an interference edge of edge (u, v).

Definition 6. For a coming request $R(s, t, B_r)$, deciding the source (s), the destination (t) and the requested bandwidth (B_r), an edge $e \in G$ is said to be a primary link if it is used for a primary path. Similarly, if e is used for a protection path, it is a protection link. Otherwise, e is a free link.

Our **DI**verse **P**ath **RO**uting (**DIPRO**) problem can be stated in the following:

Definition 7. DIPRO problem: Given the network G with assigned channels, for a coming dynamic request $R(s, t, B_r)$, DIPRO problem seeks a pair of link-disjoint paths (a primary path P_a and a link-disjoint protection path P_b) consuming a minimum number of free links.

Note that, by providing a pair of link-disjoint paths using our proposed scheme (DIPRO) for each coming request, we can guarantee that the user request will be satisfied even with any single link failure in the network.

4.2. Diverse Path Routing Scheme

Providing diverse path scheme for a user's request is complicated, where it is affected by the number of nodes in the network and the number of available links within the network. Channel assignment also plays a big role in the availability of links, where a link exists between a pair of nodes if they have the same channel on their network interface cards.

Followed our channel assignment scheme discussed in Chapter 3, we decided the network topology and the number of available links.

To solve the diverse path routing problem, our proposed solution is based on two novel ideas which have not been well investigated in previous work. First, we embrace the interference to improve the network resource (free links) usage. After a primary path is setup, we observe that we can use links interfering with the links on the primary path, to protect the primary path. The reason lies in the fact that the primary and protection paths will never be used at the same time. Therefor, it will be resource efficient with using interfered links for protection. Second, we consider the reusability of each protection link. In other words, one protection link will be used to protect as many primary paths as possible if some criteria are satisfied. Reusing existing protection links to protect a new primary path would save the free links for any future coming requests, which could increase the network resource usage efficiency and consequently improve the network reliability.

Our scheme is listed in Algorithm 3 and 4. We represent each request Req by its source node s, its destination (target) node t and its requested bandwidth B_r. If a source node specified in a request has less than two edges, the request will be dropped since it cannot be satisfied (Lines 1-2, Algorithm 3). Otherwise, if the source node indicated in the request has at least two edges, then there might be a possibility of finding link-disjoint paths to satisfy this request. Note that, by providing each request with a pair of link-disjoint primary and protection paths, we can say that the user's request was satisfied. To satisfy user's request, First we will find a primary path (P_a) by starting with checking if our network can handle the requested bandwidth specified in the request. To indicate the links that can be used as a part of a primary path we have to do the following: *i)* remove (hide) the edges that do not have enough bandwidth to accommodate the request. *ii)* remove (hide) all primary links that have been assigned before to different requests, since any primary path (P_a) is dedicated for a specific request and cannot be shared. *iii)*

Algorithm 3: Diverse Path Routing (Step 1)

```
   input  : G, Req(s, t, B_r)
   output: Disjoint paths
 1 if Rank(s) < 2 then
 2     Drop the request;
 3 else
 4     for each edge e ∈ G do
 5         if (the residual bandwidth of e is less than B_r) OR (e is a primary or
           protection link) then
 6             Hide e;
 7     Find a shortest path P_a as the primary path for Req;
 8     if path not found then
 9         Drop the request;
10     Construct graph G' with only free and protection links;
11     for (i = 1; i < Rank(s); i++) do
12         Find a shortest path P(i);
13         // Assign a value VP(i) for path P(i) as following:
14         for each edge e on P(i) do
15             Hide e;
16             if e is interfered with P_a then
17                 VP(i) = VP(i) + R_u(e) + 1;
18             else
19                 VP(i) = VP(i) + R_u(e);
20         Add P(i) to the ProtectionSet;
21 Find_Protection(ProtectionSet);
```

remove (hide) previously protection paths used before since they cannot be assigned to serve as a part of any primary path (Lines 4-6, Algorithm 3). After removing the previous links a shortest path is found and assigned as a primary path P_a for the request Req (Line 7, Algorithm 3). Next, to continue with satisfying user's request, we need to find a link-disjoint path to form a protection path for the previously found primary path.

A shortest path algorithm can be used to find a path from the source node to the destination node specified in the request, such as Bellman-Ford algorithm [6] and Dijkstra shortest path algorithm [16].

To find a better protection path, we consider two factors. First, maintain an efficient use of network resources by reusing the protection links rather than consuming new free links. Second, embracing the network interference for a better performance, i.e., the more a link interferes with a primary path, the more preferable it is to be used for protection. To achieve these two ideas, we, in Lines 14-17 of Algorithm 3, proposed to give each path a value (VP) that depends on the amount of interference (the number of interfered links) between the protection path and the primary path, and the link reusability. Note that, each found protection path will be assigned a value VP and stored in a protection set to be used later in Algorithm 4 (Line 20, Algorithm 3).

Algorithm 4: Diverse Path Routing (Step 2)

```
   input  : ProtectionSet
   output: Protection path
 1 if ProtectionSet is not empty then
 2     for all paths in set ProtectionSet do
 3         Pick P(i) with the maximum VP(i);
 4         if P(i) has reused some protection links then
 5             for each protection link e on P(i) do
 6                 Check if the residual bandwidth of e can satisfy all the connection
                   that it handles;
 7             if all the protection links have enough residual bandwidth then
 8                 Set path P(i) as the protection path P_b for Req;
 9                 for each edge e on P_b do
10                     R_u(e) = R_u(e) + 1;
11         else
12             Set path P(i) as the protection path P_b for Req;
13             for each edge e on P_b do
14                 R_u(e) = R_u(e) + 1;
15 else
16     Drop the request;
```

After adding all possible protection paths to the protection set, we will continue with Algorithm 4, to decide which protection path will be the best candidate to be a protection

path to the previously found primary path. We in this work take into consideration the interference and the reusability when calculating the path's VP value. Therefor, the higher the VP value, the more preferable the path is to serve as a protection path. After assigning the protection path, in Line 10 and 14 of Algorithm 4, we update each link's reusability value ($R_u(e)$) by 1 each time the edge e is used as a protection link. The worst case running time for Algorithms 3 and 4 is $O(n^3m^2)$. Note that, to make sure any protection link can be reused further more and provide a protection for multiple requests, we have to check if the bandwidth on each edge $e \in P_b$ equals the bandwidth of all the requests it will protect, in other words, P_b must be able to satisfy multiple requests. (Lines 5-6, Algorithm 4).

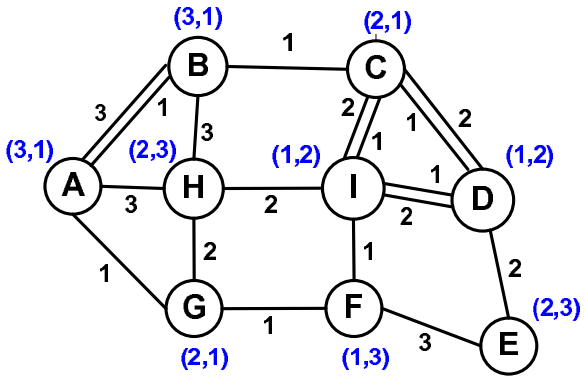


Figure 14: Channel assignment

We will use an example from Fig. 14 to Fig. 22 to illustrate our multipath solution using two requests and show how our proposed scheme satisfy the requests. Since our multipath routing scheme aims to embrace the network interference and without channel assignment it is hard to calculate the network interference. Therefor, we will use the network in Fig. 14 with channel assignment given in advance according to our channel assignment scheme discussed in Chapter 3. Each pair of nodes [(A, B), (C, D), (C, I), and (D, I)] can communicate with each other using the channels assigned on their NIC simultaneously since they share multiple channels on their NIC. This is depicted in the figures by having two links between each pair of nodes. We calculate the bandwidth on

each link e, $B(e)$, as the total channel capacity (Cap) divided by the edge interfere number of e, $IN(e)$. In short, $B(e)$ equals to $\frac{Cap}{IN(e)}$. Note that this is the best case estimation for the network in terms of bandwidth allocation fairness.

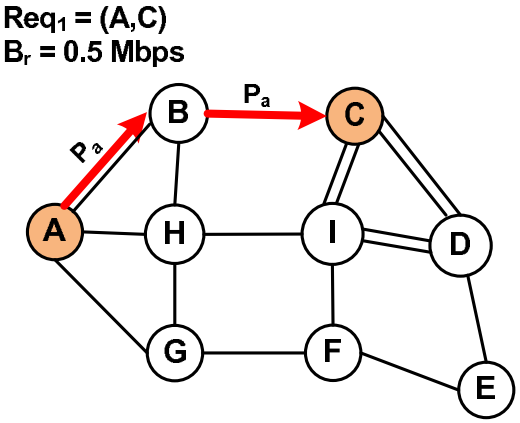


Figure 15: Request(A,C,0.5) find a primary path

Let us assume that we have two requests to be satisfied. The first request is from source node A to target node C, with a requested bandwidth of 0.5 *Mbps*. According to Algorithm 3, first we will check the source node degree to check if it has multiple links that could be used as a part in link-disjoint paths. Since node A as the source node of the first request has more that one link shared with its neighbors, we can continue with Algorithm 3. To satisfy this request, we need to find two disjoint-paths forming a pair of primary and protection paths. Using a shortest path algorithm we can find the path (A, B, C) and assign it as a primary path (P_a) for the request from node A to node C. This path is depicted with solid red arrows in Fig. 15.

To find a protection path, all the primary links used before must be removed (hidden), since they are dedicated for the primary path and they will not be used as a part of any protection path. This step is shown in Fig. 16 where the links (A, B) and (B, C) are hidden. After finding the primary path, we need to find a link-disjoint protection path to satisfy the request. By applying a shortest path algorithm [6, 16] we can find the path P_1,

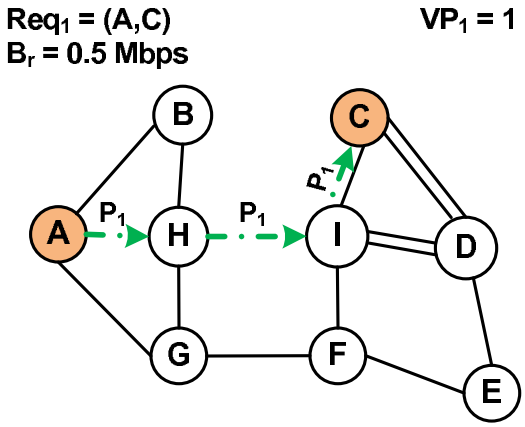


Figure 16: Request(A,C,0.5) find a protection path

(A, H, I, C). Since our proposed scheme aims to embrace the network interference and the link reusability, each path to serve as a protection path has to be assigned a value VP. Referring to our Algorithm 3 Lines 14-19 we calculate the value VP for the path P_1 to be 1.

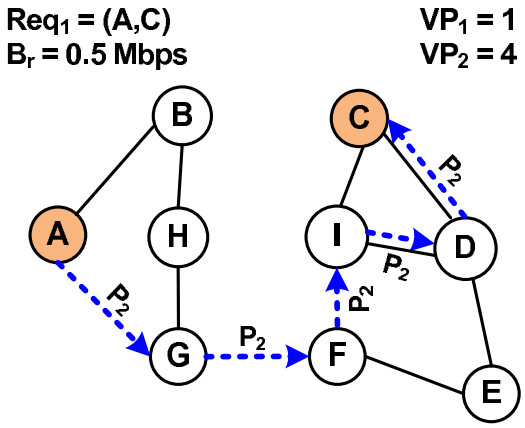


Figure 17: Request(A,C,0.5) find another protection path

After finding a possible protection path we will hide all its links and search for another possible protection path. A decision will be made according the VP values of the paths to decide on which path is the best path to serve as a protection path for the found primary path. In our example to find another possible protection path, all the perviously paths' links will not be included in the new search. Fig. 17 show all the free links after

hiding previously used links [$(A,\ B)$, $(B,\ C)$, $(A,\ H)$, and $(H,\ I)$]. Another possible protection path P_2 with edges $(A,\ G)$, $(G,\ F)$, $(F,\ I)$, $(I,\ D)$ and $(D,\ C)$ can be found by applying a shortest path algorithm such as Dijkstra or Bellman-Ford algorithms [6, 16]. The VP value of this path was calculated to be 4 according to Algorithm 3 (Line 17).

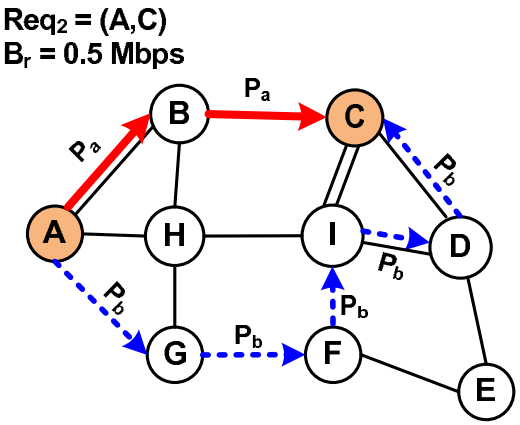


Figure 18: Pick protection path

After finding all possible protection paths and following Algorithm 4 Line 3, we will pick the path with the highest VP value, which is P_2, to serve as the protection path (P_b) for the primary path (P_a) found before. This step is shown in Fig. 18. After deciding on a protection path, a reusability factor will be updated for each link used in the protection path as given in Algorithm 4 Line 14. By assigning a pair of primary and protection paths, we can say that the request $(A,\ C,\ 0.5)$ was satisfied.

Let us consider another request (Req_2) from node H to node C with B_r as 1 *Mbps*. To satisfy this request, a pair of primary and protection link-disjoint paths need to be provided. Since primary paths are dedicated for the request and will not be shared between requests, all previously found primary paths must be hidden and not included in any new search. Moreover, the protection links used before must be hidden as well since they cannot be used as a part of any primary path. Using the free links shown in Fig. 19, a possible path can be found and assigned as a primary path P'_a for (Req_2).

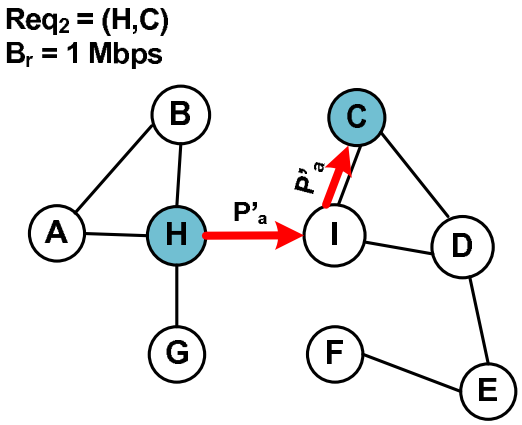


Figure 19: Request(H,C,1) find a primary path

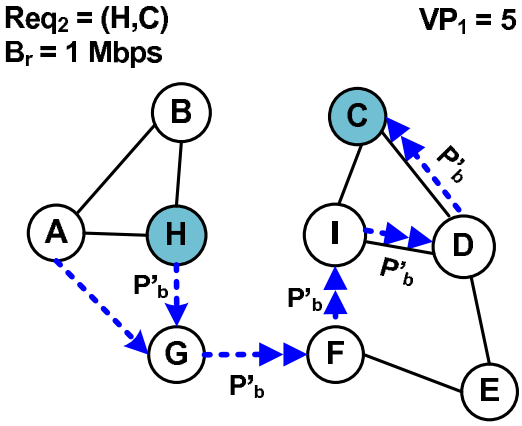


Figure 20: Request(H,C,1) find a protection path

To satisfy the coming request, a pair of primary and protection paths need to be provided. Therefor, after finding a primary path a protection path need to be found and assigned to the request. Before searching for a possible protection path, all the primary links will be hidden since they will not be included in the search. Note that previously used protection links will be included in the search for new possible protection paths, since we aim to reuse protection links to protect multiple requests (if some criteria satisfied). Using a shortest path algorithm [6, 16] a possible protection path P_1 = (H, G, F, I, D, C) can be found with a VP value calculated to be 5 as given in Fig. 20. The double arrows in the figure indicates the link reusability, where these edges have been reused for protection.

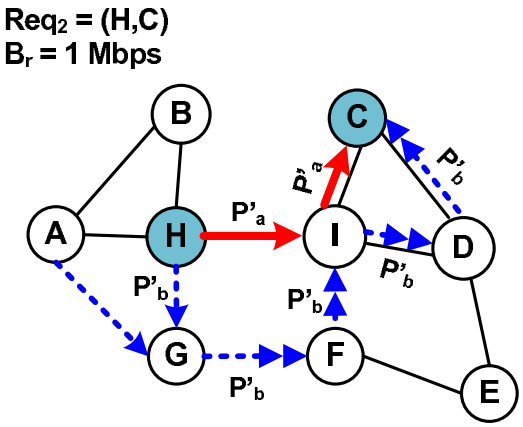


Figure 21: Satisfy both requests

To find another possible protection path, we need to hide all the links forming the previous possible path P_1 to avoid redundancy. After hiding the links it can be found that path P_1 is the only possible protection path that can serve as a protection path for the given request (Req_2). Both the primary and the protection paths for request (H, C, 1) are shown in Fig. 21.

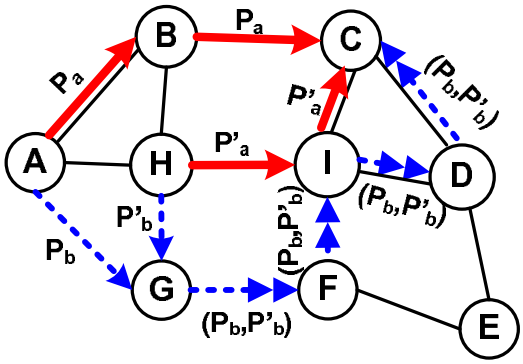


Figure 22: All primary and protection paths

By combing all the previous found paths, Fig. 22 shows that both requests were satisfied with a pair of primary and protection paths for each of them.

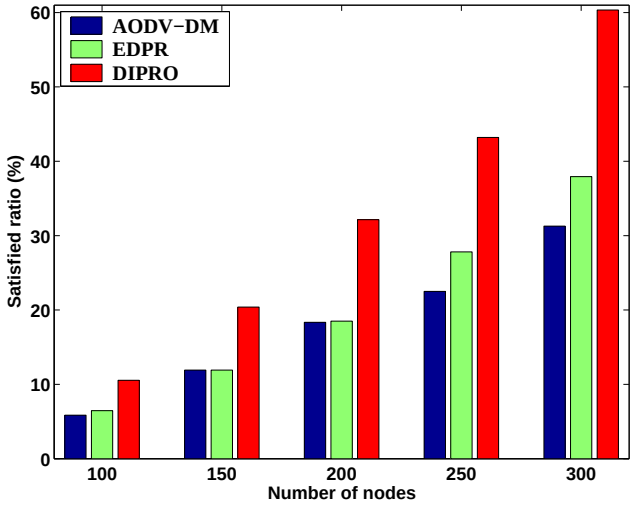
4.3. Numerical Results

In this section, to illustrate the performance of our proposed scheme, we implement our solution (denoted by **DIPRO** in the figures), and compare it to the previous schemes in [57] (denoted by **EDPR** in the figures) and [28] (denoted by **AODV-DM** in the figures). As in [28], we considered static WMNs with n nodes uniformly distributed in a square playing field. Each node has a 250 meters fixed transmission range, and a 500 meters interference range [50]. Our simulations are realized using LEDA 4.2 [25]. The results shown are the average of 10 test runs for various scenarios. All the requests in the experiments were generated randomly.

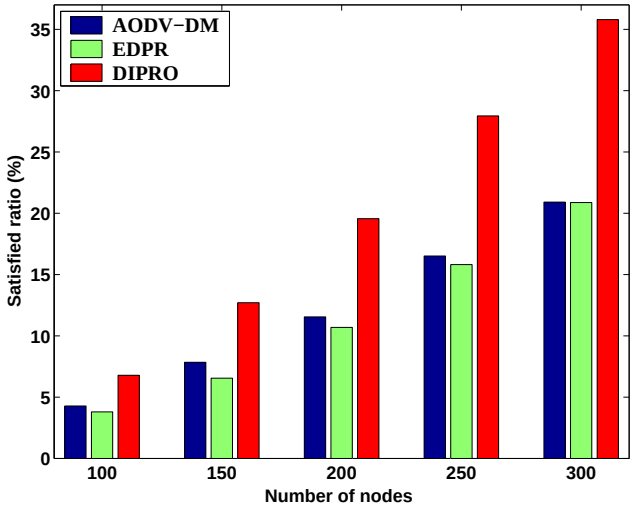
The first metric we use for performance evaluation is the **satisfied ratio**. By providing a pair of link-disjoint paths for a request, it is said that this request was satisfied. satisfied ratio of a scheme was calculated as the number of satisfied requests divided by the total number of requests. Our second performance metric we consider in this evaluation is the **running time**, which define as the time a scheme takes to process all the requests. We tested the performance with different network densities, where density is the number of nodes in one square unit.

We present our numerical results into two subsections. First, in subsection (4.3.1), we evaluate the performance of each scheme, assuming that when satisfying a user request, a pair of primary and protection paths will be dedicated for this specific request till the end of the simulation time. These results are denoted hereafter as (**no timeout**). In subsection (4.3.2), we evaluate the performance of our proposed scheme compared to both the previous schemes in [28] and in [57], taking into consideration users' requests life time, which is defined as the time it takes to process user's request, starting with satisfying the request till finishing the process specified in the request (*i.e*, downloading, transmitting, *etc.*). When satisfying user's request, a pair of primary and protection paths will be dedicated to each

request. After the request is over, all the dedicated paths will be released to be used later to satisfy other users' requests. These results are denoted hereafter as (**with timeout**).



(a) Satisfied ratio with 1000 requests



(b) Satisfied ratio with 2000 requests

Figure 23: Satisfied ratio with different number of nodes (no timeout)

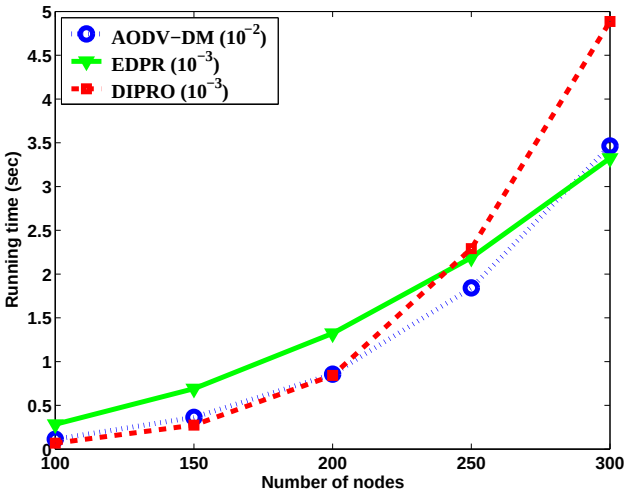
4.3.1. User satisfaction without considering requests' life time

In the first scenario, we randomly distribute different number of nodes (100, 150, 200, 250 and 300) in a 1500×1500 square meters fixed area size. We evaluate the performance of our scheme (DIPRO) and compare it to EDPR and AODV-DM schemes with different

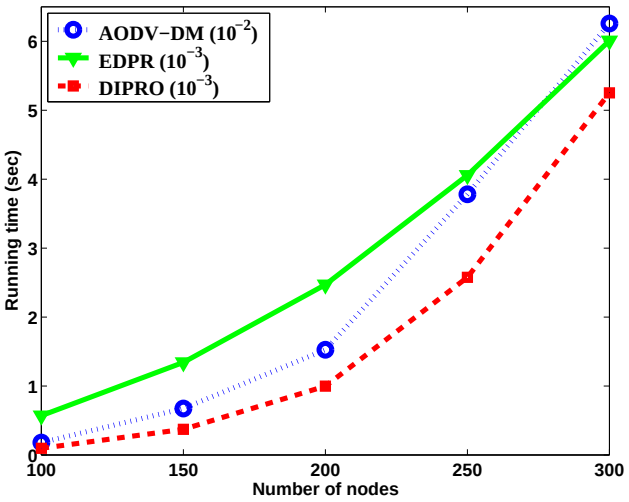
number of user requests ranges from 1000-2000. The results for our first scenario are shown in Fig. 23. The satisfied ratio of 1000 requests provided by each scheme is shown in Fig. 23(a). We observe that by increasing the density in the same area, the number of edges in the network will increase as well, due to fact that more number of nodes will be close to each other. Therefor, we can satisfy more number of requests, which is supported in the results in Fig. 23(a). For example, with 200 nodes in 1500×1500 square meters area size, our DIPRO scheme can satisfies 320 requests compared to that of 615 requests with 300 nodes in the same area size. Moreover, our results in Fig. 23(a) show that our proposed scheme (DIPRO) satisfies more requests than the other two schemes, since we tend to reuse previously chosen protection paths to protect other requests. For example, our proposed DIPRO scheme satisfies 160 more requests than EDPR scheme in a 250 nodes case. While, it satisfies 220 more requests than AODV-DM scheme under the same circumstances. Note that, in the AODV-DM scheme an insulating region has to be formed around the primary path, where a protection path will be chosen from outside of the insulating region to reduce potential network interference with the found primary path. This process could eliminate most of the links in the network, thus minimizing the ability of satisfying a large number of users' requests.

The same results' trend can be seen in Fig. 23(b), where we evaluate the performance of our scheme and compare it to previous schemes with 2000 requests. It can be seen with our proposed DIPRO scheme that by embracing interference, avoiding the use of interfered links in primary paths and allowing links reusability we can provide a better satisfied ratio compared to that with previous schemes.

The results of our first scenario in terms of the running time metric are shown in Fig. 24. The AODV decoupled multipath routing protocol (AODV-DM) required after finding a primary path to form an insulating reign around the primary path, which contains all the nodes within the interference range of each node on the primary path. A protection



(a) Running time with 1000 requests



(b) Running time with 2000 requests

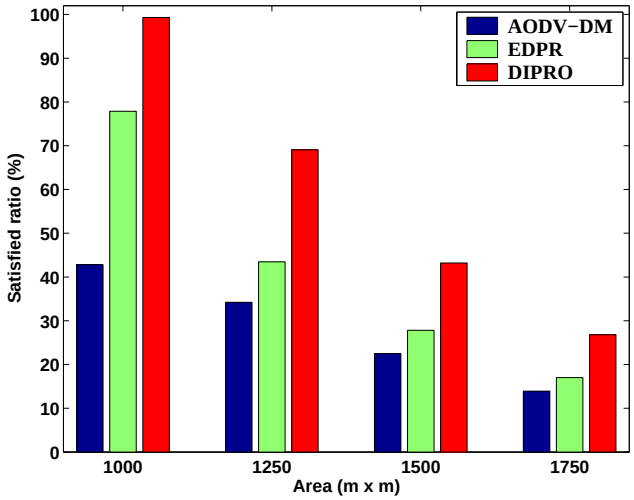
Figure 24: Running time with different number of nodes (no timeout)

path is selected outside the insulating reign to reduce potential network interference with the found primary path. We realized that the process of constructing the insulating reign consumes much time compared to the other two scheme (DIPRO and EDPR), thus we present the running time of the AODV-DM scheme with different metric notation ($\times 10^{-2}$). Fig. 24(a) presents our first scenario's results with 1000 requests in terms of running time. It can be seen that the running time increases by increasing the network density

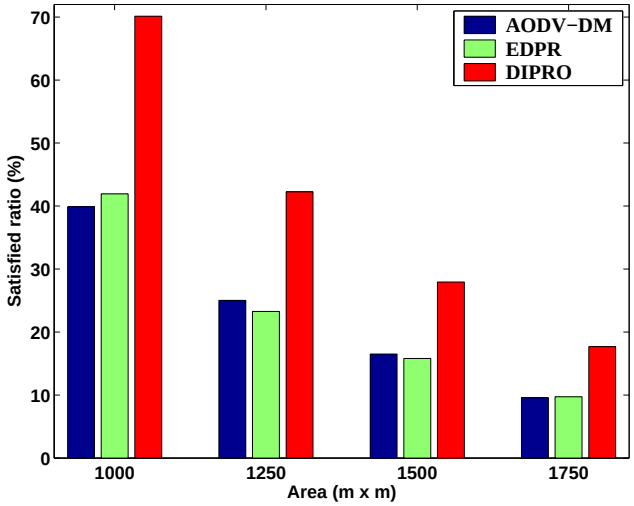
(increasing the number of nodes in a fixed area size), through the increase in the network links, and consequently increases the search time in finding the best link-disjoint paths to satisfy users' requests. Since our DIPRO scheme aims to reuse the protection paths (if some criteria satisfied), it can save time in searching for a protection path, where it can be seen that in most cases, the time it takes for our proposed DIPRO scheme to satisfy users' requests outperforms the two previous schemes. For example, with 250 nodes, our proposed scheme takes ($1.62 \times 10^{-3} sec$) to satisfy 430 users' requests, compared to that of ($2.23 \times 10^{-3} sec$) to satisfy 279 users' requests using EDPR scheme, while it takes the AODV-DM scheme ($2.34 \times 10^{-2} sec$) to satisfy 230 users' requests. Fig. 24(b) represents our first scenario's results in terms of running time with 2000 users' requests. The same result's trend can be seen in Fig. 24(b), where increasing the network density will result in increasing the time consumed in satisfying users' requests, due to the increase in network links, which increases the chance of finding more link-disjoint paths to satisfy larger number of requests. It can be seen that our scheme also outperforms previous schemes in terms of running time.

To show the effect of network density (with different area sizes) on the the performance of our proposed scheme compared to the previous schemes in terms of the satisfied ratio and the running time metrics, we in our second scenario distribute 250 nodes in four different area sizes ($100 \times 10^4, 156.25 \times 10^4, 225 \times 10^4, 306.25 \times 10^4$) square meters. The corresponding results for the second scenario in terms of satisfied ratio and the running time are shown in Fig. 25 and Fig. 26, respectively. We notice that, by increasing the area size, while keeping the number of nodes fixed, the satisfied ratio decreases. In dense networks, nodes will be close to each other and more number of links will be available, which allows us to satisfy more number of users by providing each request with a pair of primary and protection paths. On the other hand, in sparse network, nodes will be away from each other

and less number of links will be available to satisfy user requests. This was supported by our results in Fig. 25 with a drop in the satisfied ratio through the increase in the area size.



(a) Satisfied ratio with 1000 requests



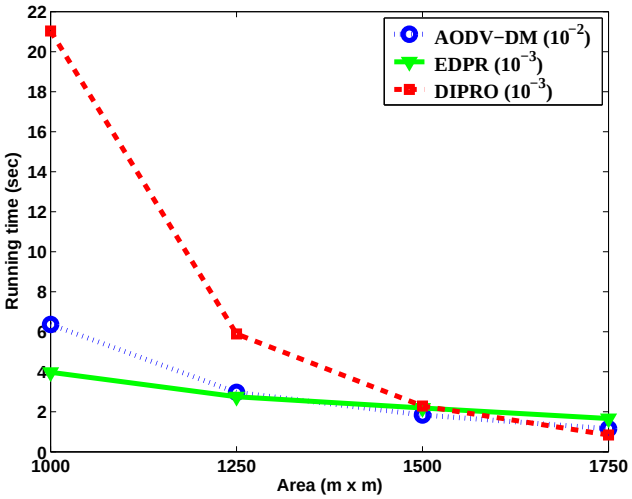
(b) Satisfied ratio with 2000 requests

Figure 25: Satisfied ratio with different area sizes (no timeout)

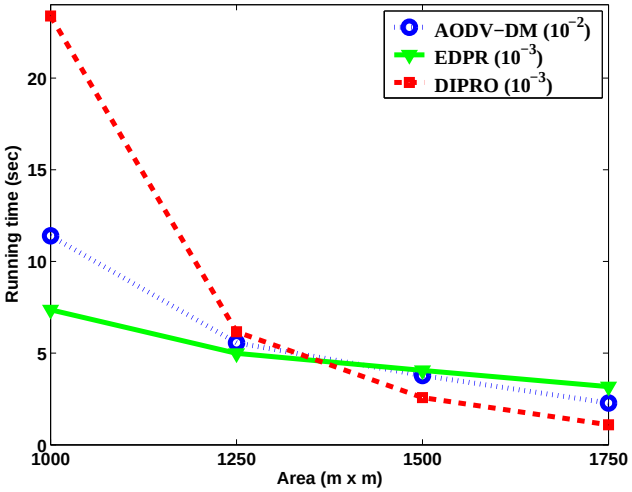
Fig. 25(a) showed the satisfied ratio provided by each scheme, with the distribution of 250 nodes in four different area sizes. Here, we tested this scenario by providing 1000 users' requests. The results show that our DIPRO scheme satisfies more requests than the other two schemes. For example, DIPRO satisfies 160 more requests than EDPR scheme in

the 1000×1000 square meters area size. While it satisfies 530 more requests compared to that with AODV-DM scheme. It also can be seen that, in sparse network the satisfied ratio decreases due to the decrease in the number of available links. But it is still obvious that our DIPRO scheme provides a better satisfied ratio compared to the other two schemes. Moreover, We tested our second scenario with 2000 requests and the corresponding results are shown in Fig. 25(b). It can be seen that, our proposed DIPRO scheme outperforms the other two schemes, where more number of requests are satisfied. For example, in Fig. 25(b) with 1000×1000 square meters area size, our DIPRO scheme satisfies 1460 requests out of 2000 requests, while EDPR scheme satisfies 1050 requests out of 2000 requests and AODV-DM scheme satisfies 824 requests out of 2000 requests.

The results for our second scenario in terms of running time are shown in Fig. 26. Our results show a decrease in the running time with the decrease of network density. Note that, in sparse networks, nodes could be away from each other, which results in less number of links to satisfy users' requests, thus having short running time to process all users' requests. We tested our second scenario with 1000 requests, and present the corresponding results in Fig. 26(a). It can be seen that in most cases, our DIPRO scheme outperforms the other two schemes in terms of running time. For example, in 1000×1000 square meters area size, our proposed DIPRO scheme satisfies 986 requests out of 1000 requests in $6.12 \times 10^{-3} sec$, while EDPR scheme satisfies 762 requests out of 1000 requests in $4.22 \times 10^{-3} sec$. On the other hand, AODV-DM scheme consumes much time forming the insulating region and picking the protection path outside the insulating region to reduce potential network interference with the found primary path. This is shown in Fig. 26(a) by having higher running time. Note that, we use different metric notation to show the running time of AODV-DM scheme. To support our results, we test our second scenario with 2000 requests and present the results in Fig. 26(b). With more number of request, it is more clear that



(a) Running time with 1000 requests



(b) Running time with 2000 requests

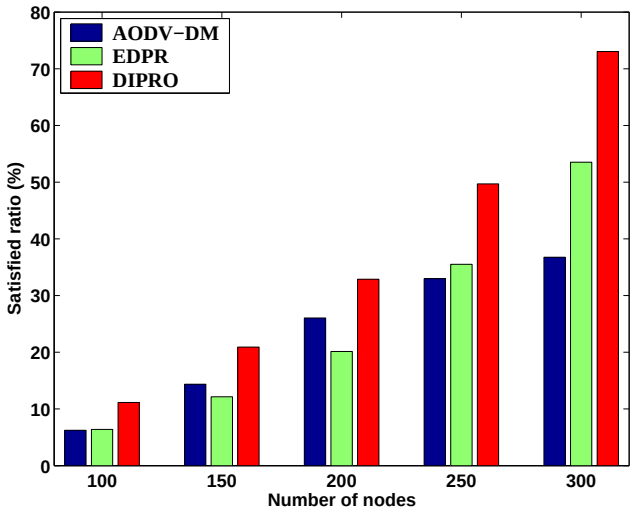
Figure 26: Running time with different area sizes (no timeout)

our DIPRO scheme provides a better satisfied ratio with a shorter running time compared to the other two schemes.

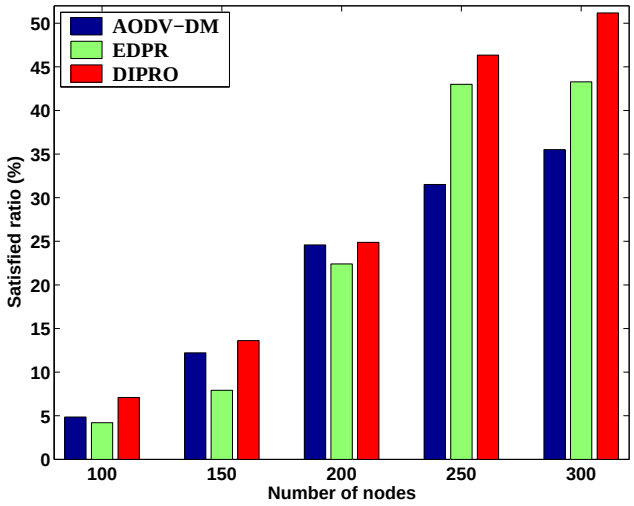
4.3.2. User satisfaction with life time consideration

To illustrate the performance of our proposed scheme (DIPRO) taking into consideration the request life time, we implement our scheme and compare it to EDPR and AODV-DM schemes. We define the request life time as the total time the request takes from entering the network till the time it is done, which includes the time to satisfy the request. We evaluate the performance of our DIPRO scheme and compare it to the other two schemes in two different scenarios. In our first scenario, different number of nodes (100, 150, 200, 250 and 300) are distributed in a 1500×1500 square meters area size. We test the performance of the schemes with 1000 and 2000 requests and the corresponding results are shown in Fig. 27. The results of our first test case are shown in Fig. 27(a), where we test the performance of each scheme by providing 1000 requests and evaluated how many requests each scheme can satisfy (provide a pair of primary and protection paths per request). Note that, previously, when satisfying a user's request, a pair of primary and protection paths will be dedicated to that request. It is worth nothing that by considering the request's life time and after the request timed out, all the paths previously dedicated to a specific request will be released and used later to satisfy another request, which provides a better satisfied ratio in each scheme.

It can be seen from Fig. 27(a) that by increasing the number of nodes in a specific area size, the satisfied ratio provided by each scheme will increase as well, due to the increase in the available links. Our results in Fig. 27(a) show that our proposed scheme outperforms previous schemes in term of providing a higher satisfied ratio. For example, with 300 nodes distributed in a 1500×1500 square meters area size our DIPRO scheme satisfies 725 requests out of 1000 requests compared to that of 523 requests with EDPR scheme and 374 requests with AODV-DM scheme. To support our results, we test the first scenario with 2000 requests and present the results in Fig. 27(b). The same results' trend can be seen from the figures. First, we realized that increasing the number of nodes in the



(a) Satisfied ratio with 1000 requests

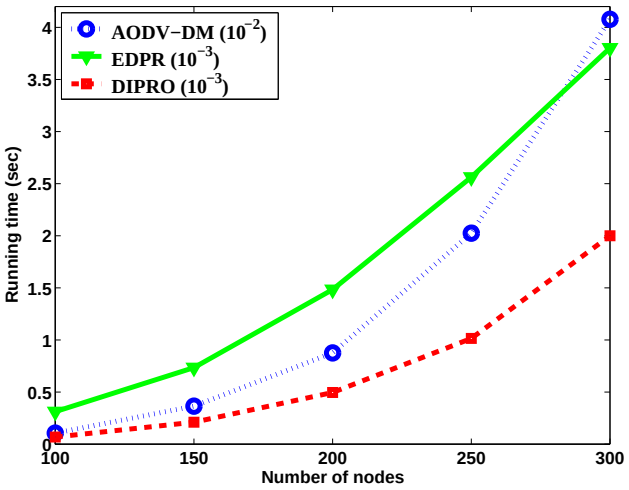


(b) Satisfied ratio with 2000 requests

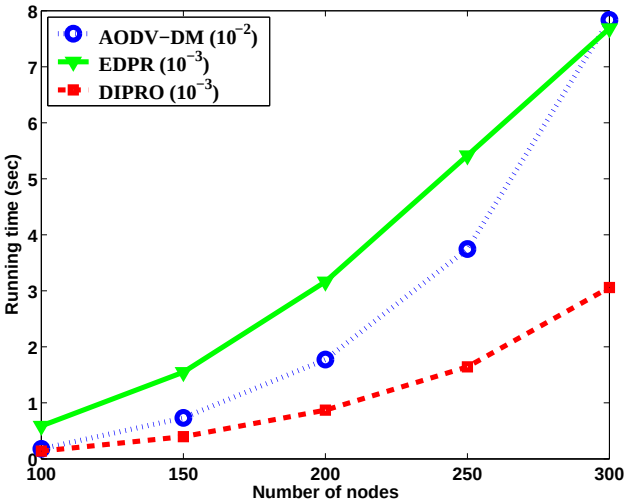
Figure 27: Satisfied ratio with different number of nodes (with timeout)

same area size, will increase the network density, due to the increase in the available links, and this will result in increasing the satisfied ratio provided by each scheme. Second, our results showed that by applying our proposed scheme we are able to satisfy more number of requests compared to that with the other two schemes.

Fig. 28 showed the performance of the three schemes in terms of running time. Our results show that by increasing the network density, each scheme is able to satisfy more



(a) Running time with 1000 requests

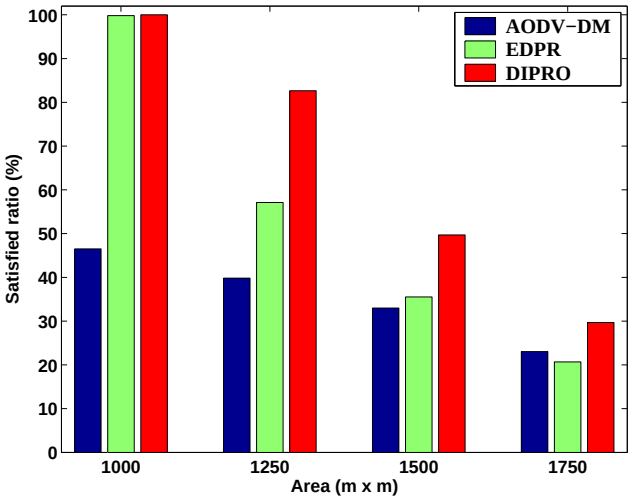


(b) Running time with 2000 requests

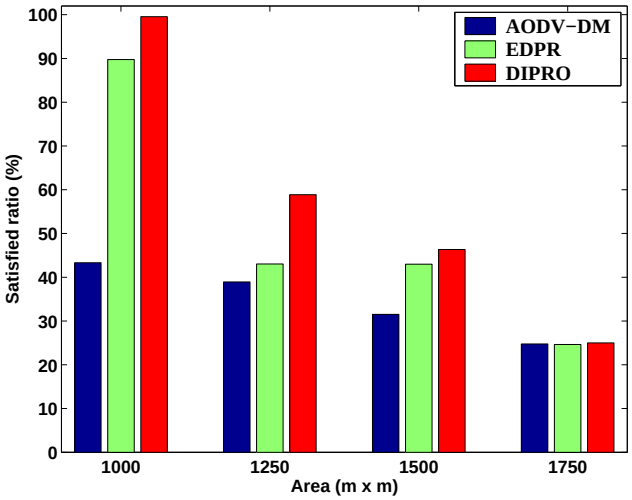
Figure 28: Running time with different number of nodes (with timeout)

number of users' requests, due to having more available free links in the network. And this is reflected in consuming more running time in denser networks. For example, in Fig. 28(a) with 200 nodes, our DIPRO scheme satisfies 323 requests out of 1000 requests in $0.5 \times 10^{-3} sec$, while it will satisfy 734 requests out of 1000 requests in $2.1 \times 10^{-3} sec$ in 300 nodes case. Compared to the other two scheme, our proposed DIPRO scheme satisfies more number of requests in a shorter period of time. For example, with 250 nodes case, it

can be seen that DIPRO scheme satisfies 502 requests out of 1000 requests in 1.1×10^{-3}sec, while EDPR scheme satisfies 345 requests in $2.3 \times 10^{-3} sec$ and AODV-DM satisfies 320 requests in $2.7 \times 10^{-2} sec$. To support out previous results, we test our first scenario with larger number of users' requests (2000 requests). The corresponding results are shown in Fig. 28(b), which presents the same results' trend.



(a) Satisfied ratio with 1000 requests



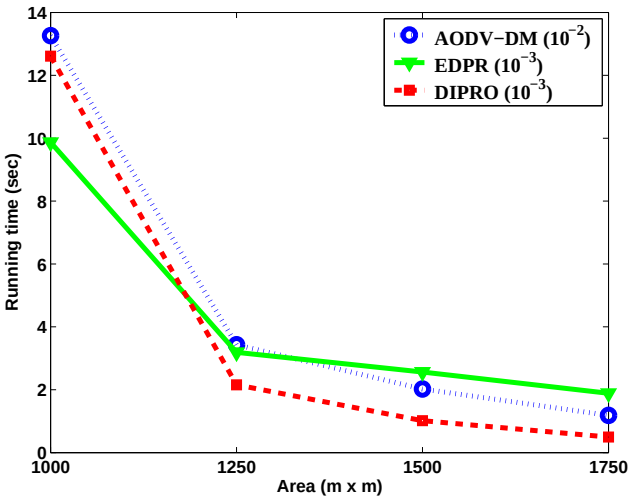
(b) Satisfied ratio with 2000 requests

Figure 29: Satisfied ratio with different area sizes (with timeout)

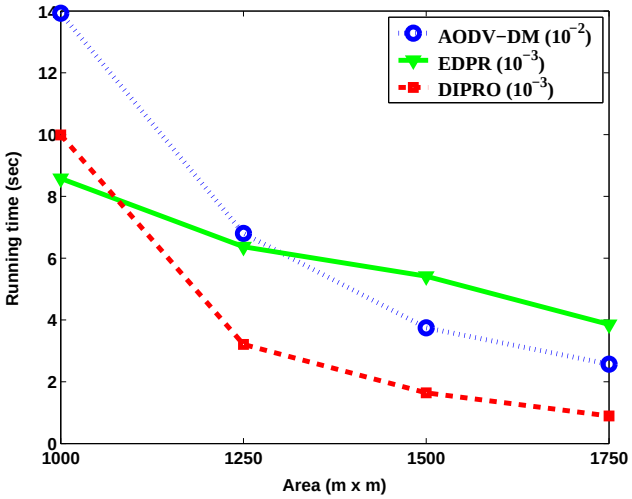
To show the impact of the area size on the performance of our scheme, we, in our second scenario, distribute 250 nodes in four different area sizes and evaluate and compared our scheme to the previous schemes (EDPR and AODV-DM) in terms of satisfied ratio and running time. Our results for the satisfied ratio are shown in Fig. 29. We realize that the satisfied ratio decreases with the decrease in network density, due to having less number of available links to satisfy users' requests caused by the nodes being away from each other. We test our second scenario with 1000 and 2000 requests, and we present the corresponding results in Fig. 29(a) and Fig. 29(b), respectively. It can be seen that our DIPRO scheme provides a better satisfied ratio, through avoiding the use of interfered links in different primary paths as well as the reusability of the protection links, compared to that with EDPR and AODV-DM schemes. For example, in Fig. 29(b) in 1250×1250 square meters area size, our DIPRO scheme provides 17.2% more satisfied ratio compared to the EDPR scheme, and it provides 20% more satisfied ratio compared to the AODV-DM scheme.

Our results for the running time metric are presented in Fig. 30. Note that, in sparse network, nodes will be away from each other, which will decrease the number of links in the networks. Our previous results in Fig. 29 show that the number of satisfied requests will decrease by the decrease in the network density (increase the area size), due to the decrease in the number of available links, which is reflected in having shorter running times to process coming requests. Our results showed that DIPRO scheme provides a shorter running time compared to the other two scheme, due to the reuse of protection links to satisfy other users' requests. Fig. 30(a) show the running time to satisfy 1000 requests. It can be seen that our DIPRO scheme provide a better running time compared to the other two scheme. For example, in 1500×1500 square meters area size, our DIPRO scheme provides a $1.7 \times 10^{-3} sec$ running time compared to that of $2.1 \times 10^{-3} sec$ when using the

EDPR scheme and $2.9 \times 10^{-2} sec$ with the use of AODV-DM scheme. The same results' trend can be seen in Fig. 30(b), where we test our second scenario with 2000 requests.



(a) Running time with 1000 requests

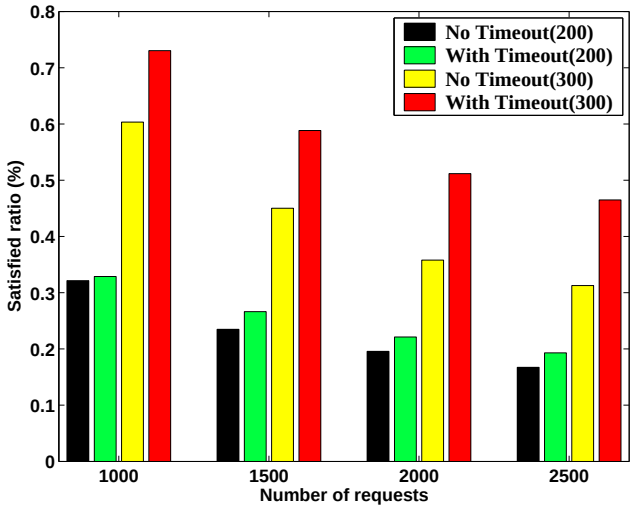


(b) Running time with 2000 requests

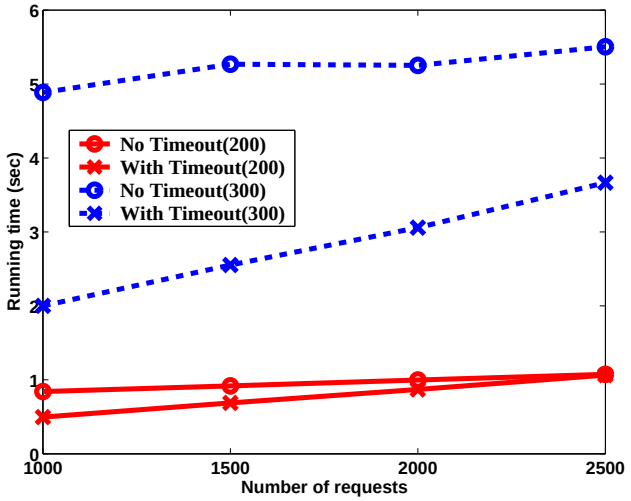
Figure 30: Running time with different area sizes (with timeout)

To show the performance of our scheme with different number of requests, we distribute 200 and 300 nodes in 1500×1500 square meters area size and show the performance of our DIPRO scheme when considering the requests' life time and compare it with that without considering the requests' life time (when satisfying users' requests, the pair of

primary and protection paths will be dedicated to that requests throughout the simulation time).



(a) Satisfied ratio with different number of requests



(b) Running time with different number of requests

Figure 31: Performance comparisons with different number of requests

Fig. 31(a) show the performance of our DIPRO scheme in terms of satisfied ratio. First, it can be seen that by considering requests' life time we are able to satisfy more number of requests compared to that without considering the requests' life lime. For example, with 300 we satisfy 207 more number of requests out of 1500 requests when

requests' life time was taken into consideration. Second, we notice that increasing the number of nodes while keeping the area size fixed, will increase the satisfied ratio, due to the increase in the network density, and the increase in the availability of more number of links to satisfy users' requests. For example, with 200 nodes, our scheme provides 21% satisfied ratio out of 2000 requests compared to that of 52% satisfied ratio out of 2000 requests with 300 nodes.

We, in Fig. 31(b), present the performance of our DIPRO scheme in terms of running time, where we evaluate its performance taking into consideration the requests' life time and compare it to that without the life time consideration. First, it can be seen that increasing the number of requests in the network will increase the time it takes for the scheme to satisfy users' requests. Second, we notice that increasing the number of nodes in a fixed area size will decrease the running time of our DIPRO scheme. Since our DIPRO scheme aims to provide a reusability of the protection paths (a protection path can be used further more to satisfy different users' requests, if some criteria satisfied) the time it takes for the scheme to search for new protection paths will be reduced, where instead of searching for new links it can reuse previously found protection links. Also we realize that increasing the number of nodes in the network will increase the number of available links to be used in satisfying users' requests, which also results in shorten the running time of our scheme. Finally, by considering the requests' life time, all the links previously dedicated to satisfy the user request will be released and used afterward to satisfy another user's request. This increase in the number of available links in the network will shorten the running time of the scheme as well.

CHAPTER 5. SECURE KEY MANAGEMENT

Wireless communications offer users and organizations many benefits including portability, flexibility, productivity increase, ease of installation and low cost. With more attentions on wireless mesh networks (WMNs) lately, the security issues become more important and urgent for managing and deploying in such networks [72]. The flexible deployment nature and the lack of fixed infrastructure make WMNs suffer from a variety of security attacks holding back the potential advantages and wide scale deployment of this promising wireless network technology [23, 72]. Most current security mechanisms (*e.g.*, encryption and digital signature) which can be used in WMNs are based on cryptographic keys and thus providing a well designed key management services are in demand which are responsible for establishing a trusted secure communication between nodes and keeping track of bindings between keys [4, 23]. User authentication, privacy and integrity are some examples of security requirements that can be addressed by building upon a solid key management framework [23, 38].

The way in which encryption keys are distributed among nodes in the network has an impact on making the network resistant or vulnerable to malicious attacks. Wireless networks are vulnerable to passive and active malicious attacks, where in active attacks such as the black/grey hole attacks [1, 3, 20, 55] the adversary is able to manipulate networks' packets after compromising an arbitrary node. Passive attacks are hard to detect, since the adversary after compromising a node will act normal and will just listen to all the transmissions in its range without being detected [68]. In this chapter, we propose a secure

The material in this chapter was co-authored by Farah Kandah (North Dakota State University), Weiyi Zhang (AT&T Research Labs), Xiaojiang Du (Temple University) and Yashaswi Singh (North Dakota State University). Farah Kandah had primary responsibility for the algorithm design, the implementation an the extraction of the simulation results. The original work in this chapter was presented and published at the IEEE International Conference on Communications (ICC 2011) [32]. Farah Kandah was the primary developer of the conclusions that are provided here. Farah Kandah also drafted and revised all versions of this chapter. Weiyi Zhang, Xiaojiang Du and Yashaswi Singh served as proofreaders and checked the logic and the math in the design conducted by Farah Kandah.

key management scheme seeking an encryption key assignment to mitigate eavesdropping malicious attacks.

The rest of this chapter is organized as follows. Our system model is discussed in Section 5.1. We formally define our problem we are going to study in Section 5.2. Our secure key management scheme is presented in Section 5.3, which is followed by the numerical results in Section 5.4.

5.1. Network and Threat Models

First in this section, we will describe our network model. Then, formally we will discuss the adversary (threat) model presented in this work. Note that, in this chapter the terms edge and link are interchangeable, the terms mesh router (MR) and mesh node or simply node are interchangeable, and the terms encryption key and key are used interchangeably hereafter.

5.1.1. Network Model

We assume a large WMN consists of a number of mesh routers (MR)s which are stationary and without energy constraints. These MRs provide access to mesh clients and also relay information from one MR to another through wireless multi-hop. All MRs use the same fixed transmission power ($r > 0$). We use an undirected bi-connected graph $G(V, E)$ to model the wireless mesh network where V is the set of n nodes and E is the set of m links in the network. For each pair of nodes (u, v), there exist an undirected edge $e \in E$ if and only if $d(u, v) \leq r$, where $d(u, v)$ is the Euclidean distance between u and v, and r is the transmission range of node u. Each edge between any pair of nodes (u, v) in G corresponds to a potential wireless link between nodes u and v in the network. Note that, in this work for the sake of security and to enforce a secure communication among the network, we assume that there is no communication between any two neighboring nodes (nodes in the transmission range of each other) unless they shared a common encryption key ($keys(u) \cap keys(v) \neq \varnothing$).

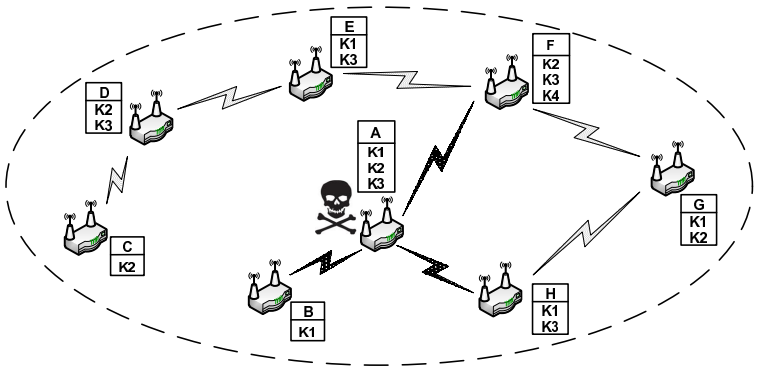


Figure 32: Malicious eavesdropping attack

5.1.2. Threat Model

To disturb WMN operations, the adversary may launch arbitrary attacks such as passive eavesdropping, bogus message injection and physical-layer jamming [54]. In this chapter we focus on passive eavesdropping attacks in WMNs. Due to the broadcast nature of wireless channels, all nodes that are in the transmission range of a specific node, say u, can receive its transmitted messages [64]. In this work we assume that the adversary can compromise an arbitrary number of mesh nodes, through physical capture or software bugs, thus gaining full control of them. Once compromised, the adversary will extract all the security information stored in the compromised nodes and all the encryption keys preloaded into their memories. Since the adversary has the ability to capture any message sent by any of the compromised node's neighbors, it will be able to decrypt any message and extract its content, if a message was encrypted using any encryption key preloaded to the compromised node memory.

To illustrate the malicious eavesdropping attack, we will use an example in Fig. 32. In this example, 8 MRs are forming a WMN where each MR is preloaded with a number of encryption keys. Note that, each node can communicate with its neighbor if they shared at least one encryption key, where the links in the figure correspond to the existence of shared keys between different nodes. Let us assume that node (A) was compromised by

an adversary, in this case all the one hop neighbors of node A (nodes B, F and H) will be monitored by the compromised node (A). Monitored links are denoted with black links in the figure. Let us start with node (F) sending an encrypted message to node (E). Note that, *i)* the only shard key between node F and node E is key ($K3$). *ii)* node E is 2-hop away from node A. Due to the broadcast nature of wireless networks and since node (A) is within node (F)'s transmission range and has the key ($K3$), the adversary at node (A) will listen to all the encrypted messages from node F encrypted using ($K3$), decrypt and extract their contents. The provided key assignment maintains a 1-hop secure communication among the network's nodes, but it makes the network vulnerable to the adversary's 2-hop compromise ability which we refer to it as the malicious eavesdropping ability hereafter. The same situation can happen if node (H) is broadcasting a message to (G) encrypted using key ($K1$), as long as the compromised node (A) has this key, where it will keep spying on all the messages in its neighborhood (transmission range) that are encrypted using ($K1$).

5.2. Problem Statement

In this section, we formally state the definition of our optimization problem that we are going to study.

Definition 8. Shared encryption key ($Sk_{u,v}$)**:** Given any two neighboring nodes $u, v \in G$, if there is an encryption key $k \in keys(u) \cap keys(v)$, then we can say that there exists shared encryption keys $Sk_{u,v}$ between node u and node v, where $keys(u)$ and $keys(v)$ are the sets of keys preloaded at node u and v, respectively.

Definition 9. 2-hop compromised nodes ($2CN_u$)**:** Given nodes $u, v, w \in G$, where v is a 1-hop neighbor of u, and w is a 2-hop neighbor of u via v. If node u has been compromised, then the 2-hop compromised nodes of node u ($2CN_u$) is defined as the set of nodes (w), for which node v sends messages encrypted by any key $k \in Sk_{u,v} \cap Sk_{v,w}$.

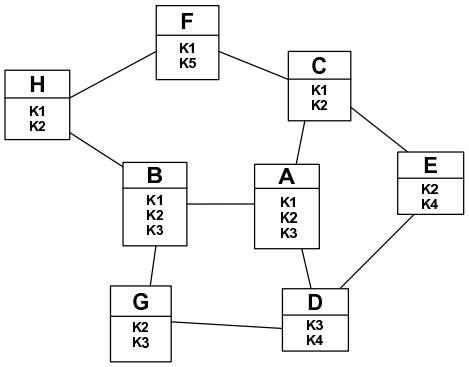


Figure 33: Key assignment (an example)

Definition 10. Node compromise ability ($NCA(u)$)**:** Given a network G, we define the node compromise ability (NCA) for a compromised node $u \in G$, as the number of nodes in the set $2CN_u$. *This is given in Eq.* 1.

$$NCA(u) = |2CN_u| \tag{1}$$

Definition 11. Malicious eavesdropping ability (MEA)**:** Given a network G with n nodes, where each node has been preloaded with a set of encryption keys. The malicious eavesdropping ability (MEA) in the network is defined as the maximum NCA among all nodes in G. *This is shown in Eq.* 2.

$$MEA = max\{NCA(n)|n \in G\} \tag{2}$$

To illustrate our definitions, we will use an example in Fig. 33 followed by all the calculations in Table. 3. In this example, we considered 8 nodes to form a wireless mesh network. Each node is preloaded with a set of keys to form a securely connected topology. Each pair of nodes will communicate if they share a common encryption key, which is depicted in the figure using solid lines between nodes.

Our calculations are shown in Table. 3. We present each node with its 1-hop neighbors, 2-hop neighbors, its 2-hop compromised node ($2CN$) list and the node compromise ability (NCA). For example, node (A) has three 1-hop neighbors (nodes B, C and D), through those nodes it will have four 2-hop neighbors (nodes E, G, F and H). Let us consider node A and node E which are 2-hop away from each other via node C. According to the definition of 2-hop compromised nodes of node A ($2CN_A$) and since there is a key $k \in keys(A, C) \cap keys(C, E)$, then node E will be considered as a 2-hop compromised node of node A. ***I.e.***, if the adversary compromises node A, it can decrypt and access all the messages sent to E from C encrypted using $K2$.

Table 3: Key assignment calculations (an example)

Nodes	***1-hop***	***2-hop***	***2CN***	***NCA***
A	B, C, D	E, G, F, H	$E_{k2}, G_{k2,k3}, F_{k1}, H_{k1,k2}$	4
B	A, G, H	C, D, F	$C_{k1,k2}, D_{k3}, F_{k1}$	3
C	A, E, F	B, D, H	$B_{k1,k2}, H_{k1}$	2
D	A, E, G	B, C	B_{k3}	1
E	C, D	A, F, G	A_{k2}	1
F	C, H	A, B, E	A_{k1}, B_{k1}	2
G	B, D	A, E, H	$A_{k2,k3}, H_{k2}$	2
H	B, F	A, C, G	$A_{k1,k2}, C_{k1}, G_{k2}$	3

Following the same procedure before, we will get that nodes G, F, and H will also be considered as 2-hop compromised nodes of node A. The node compromise ability (NCA) of node A was calculated as the number of nodes that included in the 2-hop compromised set of nodes ($2CN_A$). In our example in Fig. 33, the node compromise ability of node A was 4. By following our definitions discussed before, we can get the $2CN$ and NCA of all the nodes in the network. Finally, the malicious eavesdropping ability MEA can be calculated as the maximum NCA among all the nodes in the network. From our results in Table. 3, we can say that the malicious eavesdropping ability among the network is 4.

It is worth nothing that before encryption keys are given to each node, it is impossible to measure the malicious eavesdropping ability of the network. Note that, different encryption key assignment can induce different corresponding communications, as well as increasing or decreasing the malicious eavesdropping ability. We formalize our secure key management scheme problem in the following:

Table 4: Notation used in our scheme description

Notation	Description
u, v, w	Nodes
NIR(u)	u's neighbors that have no common keys with u
K	A set of available encryption keys
k	An encryption key
$keys(u)$	A set of keys in node u

Definition 12. SKeMS problem: Given a network G and a set of encryption keys (K), the **S**ecure **Ke**y **M**anagement **S**cheme (SKeMS) seeks a key assignment design such that the MEA in the network is minimized using $|K|$ encryption keys.

5.3. A Secure Key Management Scheme

To mitigate the malicious eavesdropping attacks in the network, we in this chapter provide a secure key management scheme (SKeMS) that seeks a key assignment scheme using K encryption keys to be assigned among all nodes in the netwrok. Our proposed solution is listed in Algorithm 5 and the notation's description to be used in our scheme is given in Table 4.

Given a network G and a set of encryption keys K, first in lines 1-2 of Algorithm 5, we initialize the set of keys ($keys$) in each node in G to an empty set. For each node in G, say u, we calculate the number of neighbors that do not share a key with u. After that we pick the node with the highest number of neighbors (*NIR*) (Lines 4-7). After choosing the node with the highest *NIR*, say u, we start assigning the keys between that node and all its neighbors (Lines 8-18). The idea in this key assignment scheme is that we assign the keys

Algorithm 5: Secure Key Management Scheme

```
   input  : G, K
   output: key management
1  for each node u ∈ G do
2      keys(u) = ∅;
3  for all nodes in G do
4      for each node u ∈ G do
5          Find NIR(u);
6          Calculate |NIR(u)|;
7      Choose node u ∈ G with the highest |NIR(u)|;
8      for each node v ∈ NIR(u) do
9          //Assign keys between node u and node v ∈ NIR(u) based on the
           following rules:
10         if keys(u) = ∅ and keys(v) = ∅ then
11             Choose k as the least used key from K;
12             Add k to keys(u) and keys(v);
13         else if keys(u) ≠ ∅ and keys(v) = ∅ then
14             Choose k as the least used key from K not in keys(w), where w is a
               neighbor of u, if applicable, else choose k as the least used key from
               K;
15             Add k to keys(u) and keys(v);
16         else if keys(u) ≠ ∅ and keys(v) ≠ ∅ then
17             Choose k as the least used key from K not on w where
               w ∈ NIR(u) ∪ NIR(v), if applicable, else choose the least used key
               from K;
18             Add k to keys(u) and keys(v);
```

among the nodes to be as different as possible, while keeping the network securely key connected. We start our key assignment scheme by taking node u and one of its neighbors, say v, as a pair of nodes and assign a key on both nodes to be used as a shared key for their secure communication. First we check to see if the chosen pair of nodes u and v has not been assigned any key yet, if so, we will choose the least used key from the set of available keys K and assign it on both nodes to be used as an encryption key for their communication (Lines 10-12 in Algorithm 5). If node u has previously been assigned some keys, in this case we will choose the least used key from K not been used on any neighboring nodes of

node u or node v, so as to assure that our key assignment will be as different as possible in the neighborhood as well as to avoid the situation of having the same common key within a 2-hop distance from node u. If our first check fail, we will check if node u has already been assigned some keys, but it is not sharing any of them with node v. In this case we will choose the least used key from the available keys not in w, where node w is a neighbor of node u, which already share a key with node u. Then we will add the key to both u and v nodes, accordingly (Algorithm 5, Lines 13-15). If the second check does not pass also, then we will check if both chosen nodes have been assigned some keys and there is no shared key between them. In this case we will choose the least used key from K not been used on either u or v's neighbors, if applicable and assign it to both nodes. otherwise, we will just choose the least used key from K and assign it to both nodes (Algorithm 5, Lines 16-18).

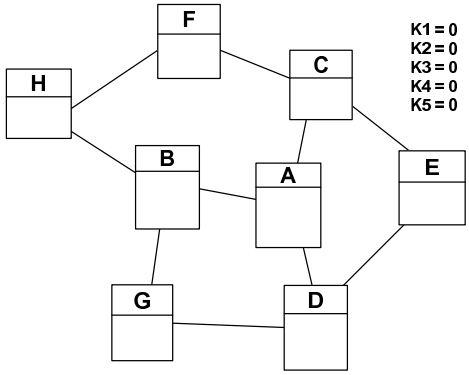


Figure 34: Key assignment example (Original topology)

We use an example from Fig. 34 to Fig. 39 to illustrate our secure key management scheme. For simplicity, we assume five keys to be assigned among 8 nodes. Fig. 34 shows the original network topology without key assignment.

According to Algorithm 5, we start with node A, since it has the highest number of neighboring nodes (including 1-hop and 2-hop neighbors) that do not share keys with it. Here we have nodes B, C and node D as node A's 1-hop neighbors. Let us start with the nodes pair (A, C), we will refer to lines 10-12 in our algorithm. We choose the least

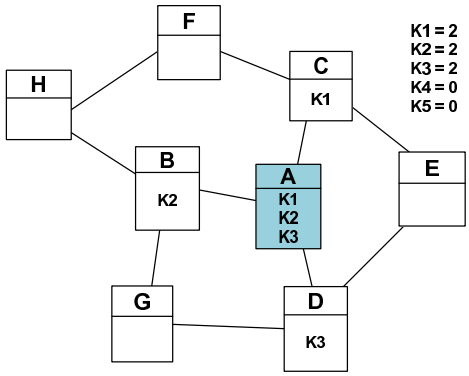


Figure 35: Key assignment example (Step 1)

used key from K, say key K1, and assign it to both nodes. Next we will follow the same previous steps to assign keys between nodes (A, B) and (A, D). This assignment can be seen in Fig. 35. Note that, a counter is maintained for each key used to keep track of the least used encryption key among the network.

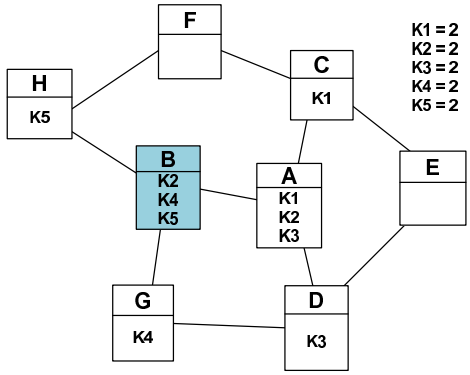


Figure 36: Key assignment example (Step 2)

Our next step is shown in Fig. 36, where we choose node B to continue the key assignment, since node B has the highest *NIR* among all the nodes in the network. It can be seen that node G and H has no shared key among them, but since node B has some keys, we will assign a new key between these two nodes to have a different key assignment to mitigate the 2-hop compromise ability of the malicious eavesdropping attack. In this case we follow our algorithm in lines 13-15 to assign the keys. Since node A which is node

B's neighbor is using keys (K1, K2, K3) for encryption, and node D which is node G's neighbor is using key K3, we will choose the least used key from K which has not been used on nodes A and D, in here we choose K4 as the encryption key between node B and node G. Following the same steps, we assign key K5 between node B and node H.

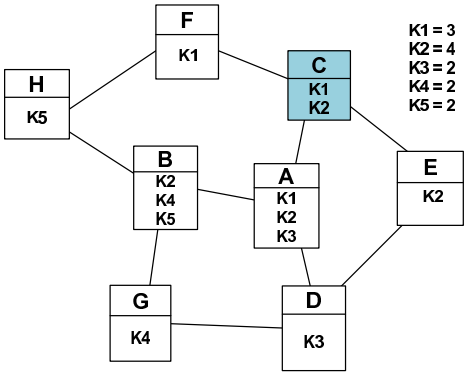


Figure 37: Key assignment example (Step 3)

After finishing with node B, we will move on to assign keys on the next node with the highest *NIR* among all the nodes in the network. Here node C and D have the same *NIR*, thus we choose the node which has the most number of neighbors with the smaller number of keys. We choose node C according to the previous rule and start assigning the encryption keys among its neighboring nodes. This step is shown in Fig. 37.

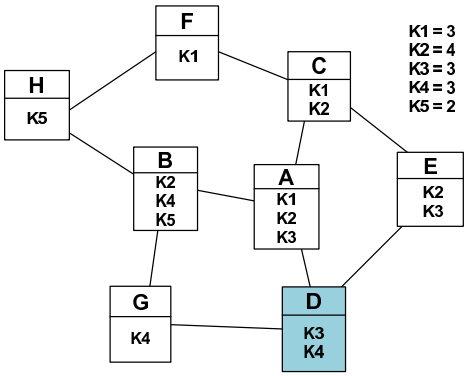


Figure 38: Key assignment example (Step 4)

Following node C, we choose node D according to Algorithm 5. The key assignment for node D is shown in Fig. 38. By applying our secure key management scheme we will get to our final key assignment shown in Fig. 39.

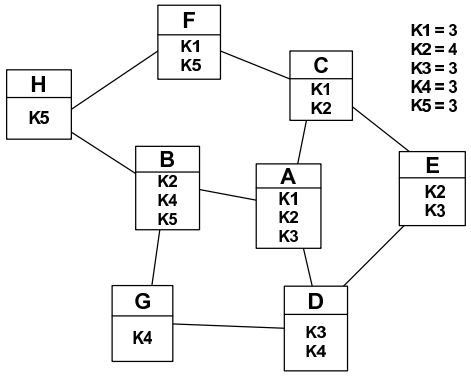


Figure 39: Key assignment example (Final)

Table 5: Secure key management scheme's calculations

Nodes	***1-hop***	***2-hop***	***2CN***	***NCA***
A	B, C, D	E, G, F, H	$E_{k2,k3}, F_{k1}$	2
B	A, G, H	C, D, F	C_{k2}, D_{k4}, F_{k5}	3
C	A, E, F	B, D, H	B_{k2}	1
D	A, E, G	B, C	B_{k4}	1
E	C, D	A, F, G	$A_{k2,k3}$	1
F	C, H	A, B, E	A_{k1}, B_{k5}	2
G	B, D	A, E, H	$-$	0
H	B, F	A, C, G	$-$	0

The corresponding calculations for our example are shown in Table. 5. It can be seen that the node compromise ability of nodes G and H equal to zero, which means that if those nodes where compromised the adversary will not be able to decrypt any message sent to nodes that are 2-hop away from the compromised node. Also our results in Table. 5 indicate that the maximum node compromise ability among all nodes in the network was 3, thus the malicious eavesdropping ability (MEA) in the network is 3, which is smaller compared to that with the key assignment provided in Fig. 33. Note that, the smaller the

(MEA) in the network, the more secure the network is against malicious eavesdropping attacks.

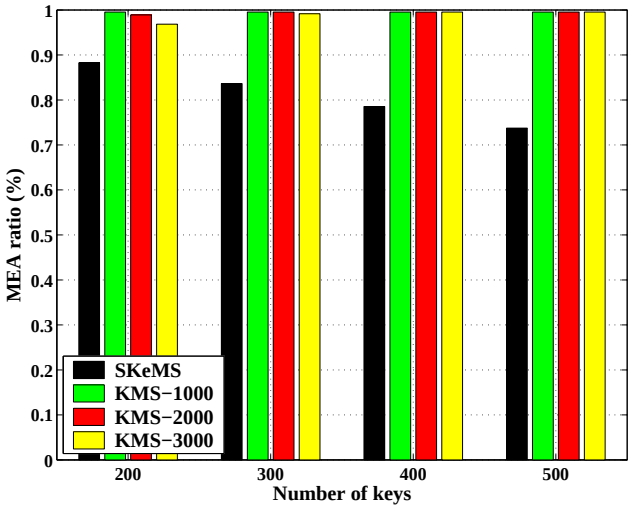
5.4. Numerical Results

To illustrate the performance of our scheme, we implemented our solution (denoted by **SKeMS** in the figures), and compared it to the previous scheme in [19] (denoted by **KMS** in the figures). We considered static WMN with n nodes uniformly distributed in a square playing field. Each node has a fixed transmission range of 250 meters. Our simulations are realized using LEDA 4.2 [25]. The results shown are an average of 5 test runs for various scenarios.

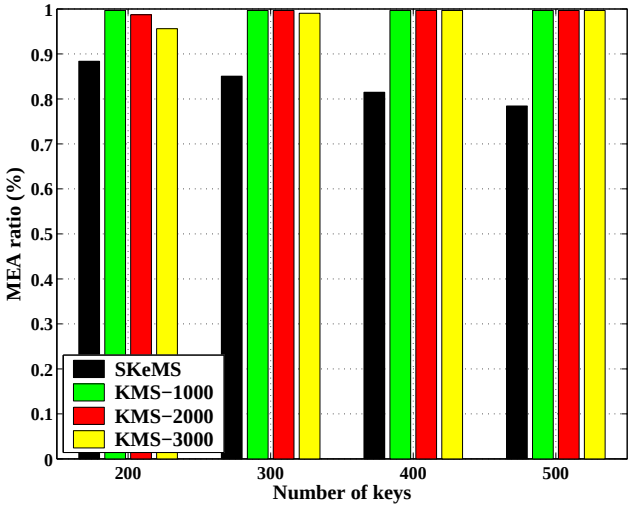
The first metric used for performance evaluation is the **malicious eavesdropping ability ratio** (denoted as MEA ratio in the figures), which is calculated as the neighbor compromise ability (NCA) divided by the total number of neighboring nodes that are vulnerable to the malicious eavesdropping attack (discussed in subsection 5.1.2). Having smaller MEA ratio indicates that the network is more secured and more resistant to malicious eavesdropping attacks.

In our first tested scenario, we randomly distributed 300 and 400 nodes in a 10×10^5 square meters. To achieve better security for KMS scheme, we provide different pool sizes range from (1000$-$3000) keys. The available number of keys K ranges from (200$-$500) keys. Note that, having different pool sizes doesn't affect our SKeMS scheme, since we distribute the set of keys (K) among all the nodes in the network rather than upload each node with the set of keys (K) as in [19]. Our first scenario's results are shown in Fig. 40.

Fig. 40(a) shows the MEA ratio versus different number of keys in the range of (200$-$500) keys. In our proposed scheme, we use the available keys to be assigned among all nodes in the network. While in the KMS, each node will be preloaded with the available number of keys which are chosen randomly from the provided pool of keys. For the KMS scheme increasing the pool size will decrease the MEA ratio. For example, with 200 keys



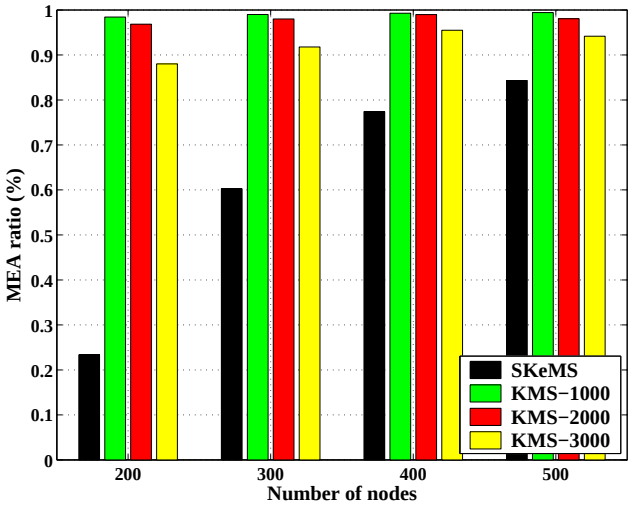
(a) 300 nodes in $1000m$x$1000m$



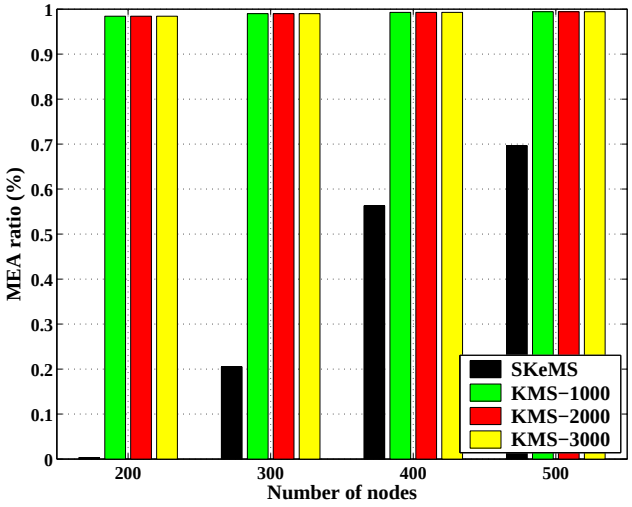
(b) 400 nodes in $1000m$x$1000m$

Figure 40: MEA ratio with different number of keys

chosen from a pool size of 1000 keys the MEA ratio is 98%, while with 3000 keys pool size with the same number of keys the MEA ratio is 94.6%. Compared to our scheme, the results show that our scheme outperforms the KMS scheme in all different tested pool sizes. For example, with 300 keys, our scheme has an MEA ratio of 82%, where by using the KMS scheme with 3000 pool size we have a MEA ratio of 98.6%. These results also show that, by increasing the number of available keys, we can provide a better MEA ratio, since we



(a) 200 Keys in a 1500mx1500m



(b) 400 Keys in a 1500mx1500m

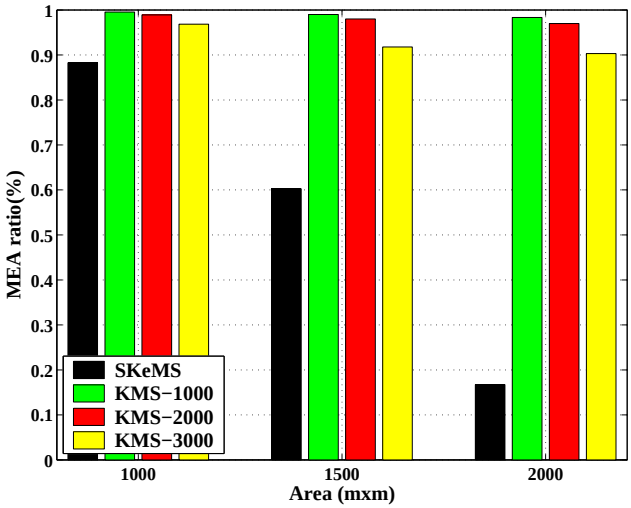
Figure 41: MEA ratio with different number of nodes

have more variety of keys to be assigned among the nodes in the network. For example, with 200 keys, the MEA ratio is 86%, while with 500 keys it drops to 73%. The same results' trend can be seen in Fig. 40(b), where we distribute 400 nodes in a 1000$\times$1000 square meters field size. Increasing network density (the number of nodes in a square area size) by increasing the number of nodes in the same area size, would increase the number of nodes that are vulnerable to malicious eavesdropping attack, since the number of neighboring nodes of the malicious nodes will increase. Our results in Fig. 40(b) show

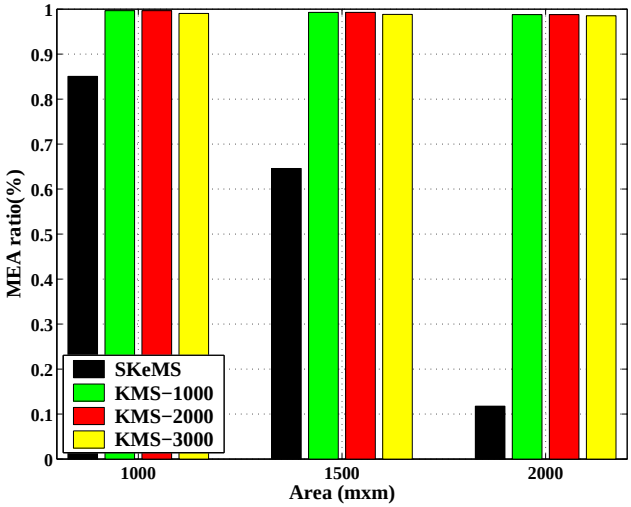
that by applying our SKeMS scheme, we can provide a better MEA ratio compared to that when applying KMS scheme with different pool sizes. For example, with 400 nodes and 200 keys, we can provide an MEA ratio of 86% compared to an MEA ratio of 96% when applying KMS scheme with 3000 pool size. Also increasing the number of keys that are available to be assigned among the nodes can provide a better security against malicious eavesdropping ability when applying our SKeMS scheme. Our results in Fig. 40(b) show that, by applying our SKeMS scheme with 200 keys we can provide an MEA ratio of 86% compared to that of 77% with 500 keys. On the other hand, KMS scheme required a large pool size to provide a better varieties of keys to be assigned to each node.

Fig. 41 show the results of our second scenario, where we studied the schemes' performance with different number of nodes (200−500) in 225×10^4 area size. In Fig. 41(a) we tested the performance with 200 available keys to be assigned in the network. The results in Fig. 41(a) show that, by applying our SKeMS scheme, the MEA ratio is increasing with the increase in the number of nodes, due to having more common shared keys between the nodes in the neighborhood. Our scheme's MEA ratio is still better compared to the MEA ratio when applying the KMS scheme. For example, in Fig. 41(a) with 200 nodes and 200 keys we can provide an MEA ratio of 24% compared to that of 87% when applying KMS scheme with 3000 pool size. The same results' trend can be seen in Fig 41(b), with 400 keys to be assigned to/among all nodes in the network.

To show the relationship between the number of available keys and the number of nodes in the network, we apply our SKeMS scheme to three different network sizes and compare it to that when applying KMS scheme. The corresponding results are shown in Fig. 42. In Fig. 42(a) we show the results of the case, where 300 nodes were distributed in different area sizes (100×10^4, 225×10^4 and 400×10^4 square meters) with 200 available keys. The results show that in sparse network, the number of neighboring nodes became smaller with increasing the area size, this indeed decreases the number of nodes that can be



(a) 300 nodes with 200 keys

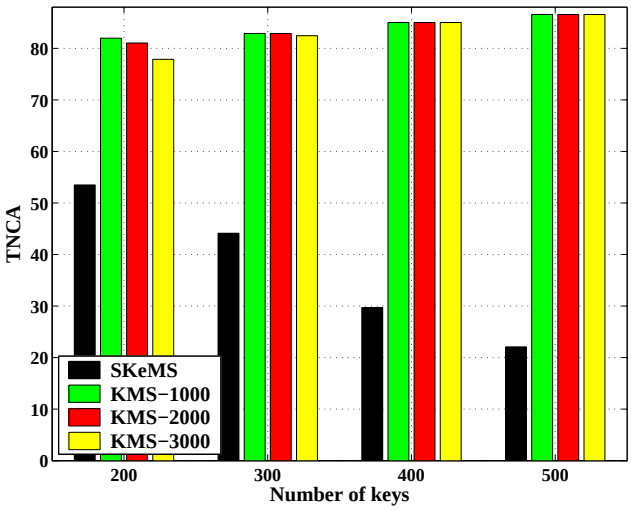


(b) 400 nodes with 300 keys

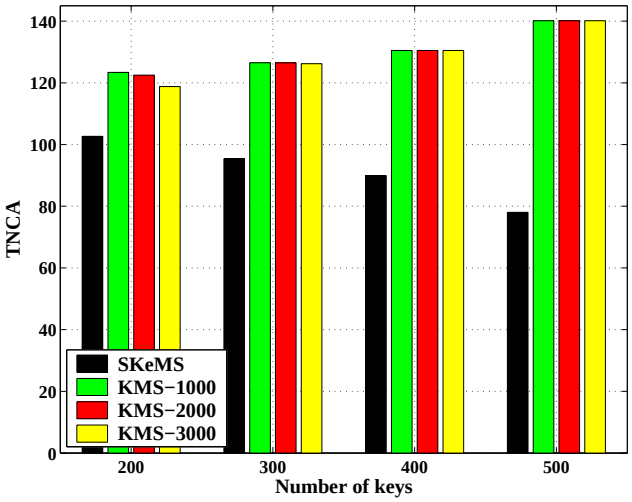
Figure 42: MEA ratio in different area sizes

affected with the malicious eavesdropping attack. For example, with our SKeMS scheme in 100×10^4 square meters area size, the MEA ratio is 86%, while by doubling the area size to 400×10^4 square meters, we can decrease the MEA ratio to 16.7%. Increasing the area size also has an effect on the KMS scheme performance. For example, with 3000 pool size and 200 keys in 100×10^4 meter square area size the MEA ratio is 98%, while this ratio decreases in a 2000×2000 square meters area size to 90%. The same trend can be seen in Fig. 42(b) with 400 nodes and 300 keys. It is obvious that our SKeMS

scheme performs much better than the previous KMS scheme in providing smaller MEA ratio which indicates a more secured network.



(a) 200 Nodes in a 1000mx1000m



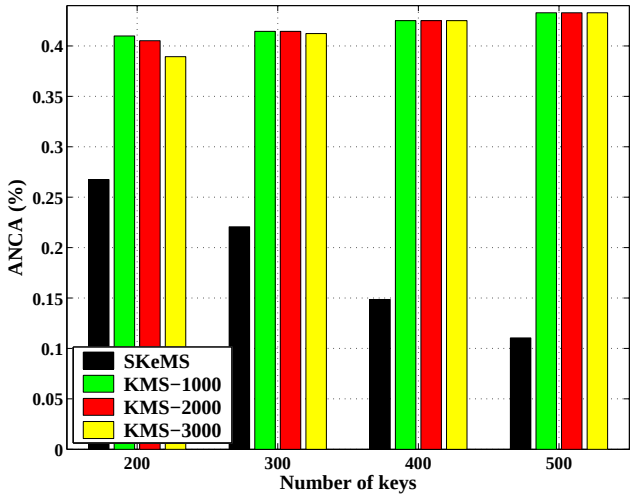
(b) 300 Nodes in a 1000mx1000m

Figure 43: Total *NCA* with different number of keys

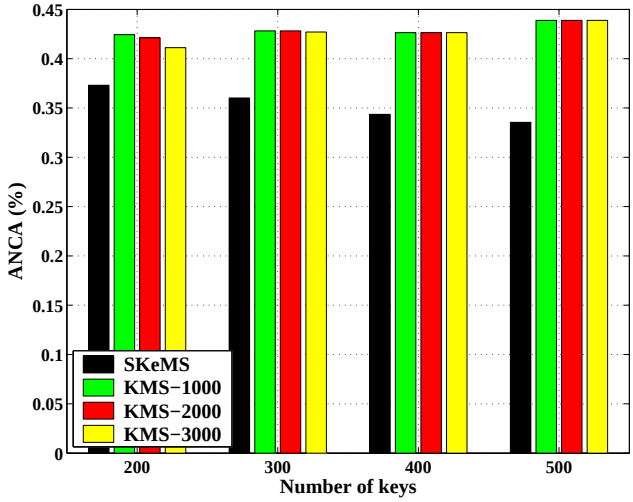
The second metric used for performance evaluation is the **total neighbor compromise ability** (denoted as TNCA in the figures), which is calculated as the total number of 2-hop nodes away from the compromised node(s) that are vulnerable to the eavesdropping malicious attack (discussed in subsection 5.1.2) launched by the malicious node(s). To show the performance of our scheme and compared it to the KMS scheme, we set the

number of malicious nodes in the network to be 10% of the total number of nodes in the networks. Our results for the total neighbor compromise ability performance metric are shown in Fig .43. In Fig. 43(a) we show the results of distributing 200 nodes in 100×10^4 square meters area size with $(200-500)$ available keys to be assigned to/among nodes in the network. Our results show that by applying our SKeMS scheme, we can decrease the total number of nodes that are vulnerable to malicious eavesdropping attack, by increasing the number of available keys to be assigned among all the nodes in the network, which provides a large variety of keys to be assigned among the nodes, while keeping the network securely connected. For example, in 200 nodes network with 200 keys, 53 nodes are vulnerable to malicious eavesdropping attack compared to that of 22 nodes when applying our SKeMS scheme with 500 keys. It is obvious that, our SKeMS scheme provide a better security among the network, by providing smaller number of nodes that are vulnerable to malicious eavesdropping attack as shown in the results in Fig. 43. To support our observation, we test our SKeMS scheme and compare it to the KMS scheme with different number of nodes under the same circumstances. Fig. 43(b) show the results of 300 nodes in 100×10^4 square area size with $(200-500)$ available keys to be assigned to/among nodes in the network. It can be seen that our SKeMS scheme provides a more secured network against the malicious eavesdropping attack (discussed in Section 5.1.2), compared to that when applying the KMS scheme.

Our third metric used for performance evaluation is the **neighbor compromise ability ratio** (denoted as ANCA in the figures), which is calculated as the total number of 2-hop nodes away from the compromised node(s) that are vulnerable to eavesdropping malicious attack (discussed in subsection 5.1.2) launched by the malicious node(s) to the total number of nodes in the networks. The smaller the ANCA ratio provided by a key management scheme, the more secure the network is. Our results for the third performance metric are shown in Fig. 44. We set the number of malicious nodes in the network to



(a) 200 Nodes in a $1000m$x$1000m$

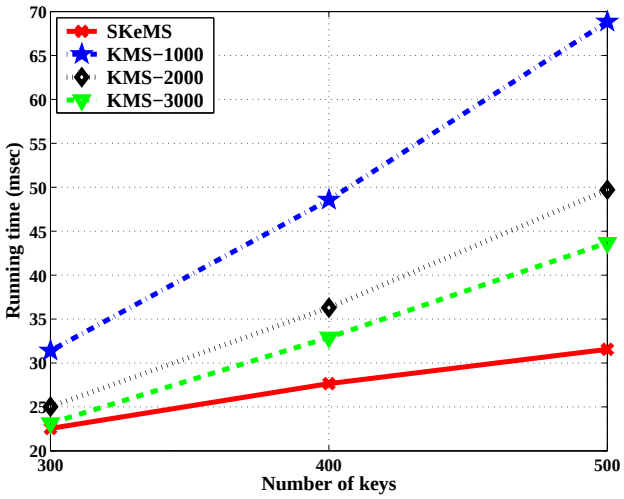


(b) 400 Nodes in a $1000m$x$1000m$

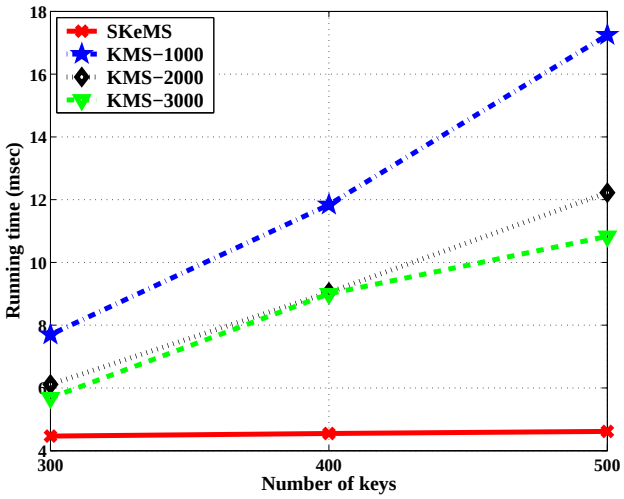
Figure 44: Average *NCA* with different number of keys

be 10% out of the total number of nodes in the network. Fig. 44(a) show the results of applying our SKeMS scheme compared to the KMS scheme, with 200 nodes distributed in 100×10^4 square meters area size. We test both key management schemes with different number of available keys ranges between (200−500 keys) to be assigned to/among the nodes in the network. It can be seen that, with our SKeMS scheme, we can provide the network with better security against malicious eavesdropping attack, indicated by smaller ANCA ratio compared to that when applying KMS scheme. For example, with 200 nodes

and 400 keys we can provide an ANCA ratio of 15% with our scheme compared to that of 42% when applying the KMS scheme. We also test the schemes performance with more dense network, where we distribute 400 nodes in 100×10^4 square meters area size. The corresponding results are shown in Fig. 44(b). The same trend can be seen in Fig. 44(b), where our scheme provides more secure network with a smaller ANCA ratio compared to that when applying the KMS scheme.



(a) 200 nodes in 1000×1000

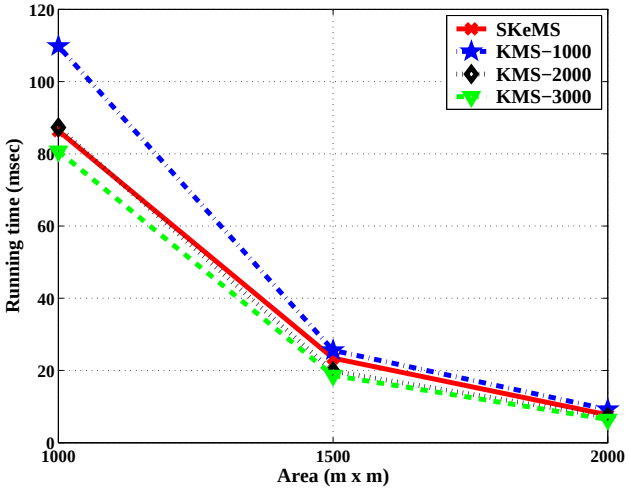


(b) 200 nodes in 1500×1500

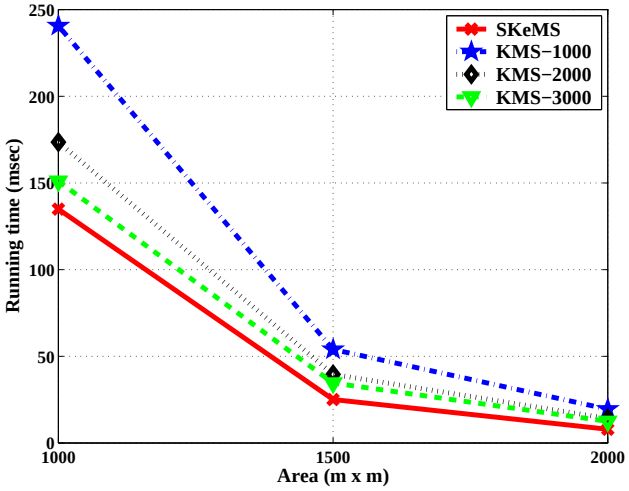
Figure 45: Running time with different number of keys

Our last performance metric used for performance evaluation is the **running time**, which is defined as the running time that the scheme takes to assign the available keys to/among the nodes in the network. We test our scheme and compare it to the KMS scheme in two different scenarios and the corresponding results are shown in Fig. 45 and Fig. 46, respectively. Our first scenario's results are is shown in Fig. 45(a), where 200 nodes were randomly distributed in a 100×10^4 square meters field size. We assign 300, 400 and 500 keys for both schemes. For the KMS scheme we measured the running time with different pool sizes ($1000-3000$). Since the KMS scheme depends on the pool size, where with different pool sizes, it can be seen that the KMS scheme takes more time in assigning the keys from 1000 keys pool size and find the MEA of the network compared to larger pool sizes, due to having more number of neighboring nodes that share more common keys. Compared to our scheme, it can be seen that our scheme outperforms the KMS scheme in all test cases. To show the impact of the network density, another case was studied and the corresponding results are shown in Fig. 45(b), where we changed the area size to be 225×10^4 square meters. It can be seen than our SKeMS scheme most of the time provides more secured network in a shorter period of time compared to that when applying the KMS scheme. For example, in Fig. 45(b), the results show that, with 400 keys, our scheme assigns the available keys among the network in 5 *msec* compared to that of 8.7 *msec* when applying the KMS scheme with 3000 pool key size and 12 *msec* with a 1000 key pool size.

The results of our second scenario in terms of the running time are shown in Fig. 46. We change the network density by increasing the area sizes from 100×10^4 to 200×10^4 square meters. Our results in Fig. 46 show that our SKeMS scheme provides a more secured network in less time compared to the KMS scheme, since it needs to randomly pick K number of different keys from the key pool and store them at each node. We test the KMS scheme with 1000, 2000 and 3000 key pool sizes. The results show that, in sparse networks the number of neighboring nodes decreases, which leads to having less



(a) 300 nodes with 300 keys



(b) 300 nodes with 500 keys

Figure 46: Running time with different area sizes

number of nodes that shared common keys in the neighborhood thus consumes less time in assigning keys to generate a secured network that is resistant to the eavesdropping attack. In Fig. 46(b) we show the performance results of our SKeMS scheme compared to the KMS scheme, when distributing 300 nodes in different area sizes. It is obvious that with 500 keys, our scheme consumes less time to provide a better secure network compared to that with the KMS scheme. For example, in 225×10^4, our SKeMS scheme consume 27 *msec*

to assign the available keys among the nodes in the network compared to that of 62 *msec* when applying the KMS scheme with 1000 keys pool size under the same circumstances.

CHAPTER 6. GENERAL CONCLUSION

In this study we investigate three design factors seeking an improvement in the performance of wireless mesh network. We define our sub-problem (1) the **I**nterference-**A**ware **R**obust **T**opology (**I-ART**) problem, and present our solution considering a network topology design and a channel assignment for the network topology to reduce the network interference while maintaining a 2-connected topology. Through simulation we show that our solution performs well in terms of network capacity, balanced ratio between maximum and minimum edge bandwidth as well as consumes less time to provide a channel assignment among the network. Our results show that our proposed scheme provides a higher network capacity in dense network compared to that using previous schemes. Moreover, we show that by assigning channels to be as even as possible in a neighborhood can provide high bandwidths among network's edges which was provided using our proposed scheme. To sum up, we realized that, by assigning the channels in a neighborhood to be as different as possible and distributed evenly among the edges in the network taking into consideration the network connectivity, the interference influence in the network can be reduced and consequently provides an improvement in the network performance.

For our sub-problem (2), we define the **DI**verse **P**ath **RO**uting (**DIPRO**) problem, and present a multipath routing scheme that provides each user's request with a pair of link-disjoint paths to satisfy the network's users and support the network reliability by avoiding any interruption in the case of any single link failure in the network. To satisfy a user request, a pair of primary and protection paths need to be assigned to the request, where the primary path will be active all the time and the protection path will be there (not active) unless there a failure occurred in the primary path. In this study, we show that any two primary paths should not use the interfered links because the interference will reduce bandwidths of both primary path, and therefor, we embraced the network interference to provide a better network performance by assigning a user request with a primary path and

choose the interfered links of that path to be a part of the protection path. Also we show that since the protection paths are not active all the time, there would be a possibility that a protection link can be a part of multiple protection paths if some criteria satisfied. This was supported by our results which show an improvement in the network performance in terms of satisfied ratio and running time.

Finally, by considering our third design factor, we defined our sub-problem (3) the **S**ecure **K**ey **M**anagement problem and present an effective solution that seeks a key assignment to provide a network that is resistant to malicious eavesdropping attacks. Simulation results showed that our solution performs well in terms of malicious eavesdropping ability ratio, total neighbor compromise ability, neighbor compromise ability ratio and less running time, compared to previously proposed schemes. We conclude that, by taking into consideration the 2-hop neighbors when assigning keys among nodes in the network we can provide a network that is resistant to malicious eavesdropping attacks.

REFERENCES

[1] P. Agrawal, R.K. Ghosh, and S.K. Das, *Cooperative black and gray hole attacks in mobile ad hoc networks*, Proceedings of the 2nd international conference on Ubiquitous information management and communication (New York, NY, USA), ICUIMC '08, ACM, 2008, pp. 310–314.

[2] I.F. Akyildiz, X. Wang, and W. Wang, *Wireless mesh networks: a survey*, Comput. Netw. ISDN Syst. **47** (2005), 445–487.

[3] M. Al-Shurman, S. Yoo, and S. Park, *Black hole attack in mobile ad hoc networks*, Proceedings of the 42nd annual Southeast regional conference (New York, NY, USA), ACM-SE 42, ACM, 2004, pp. 96–97.

[4] N. Asokan and P. Ginzboorg, *Key agreement in ad-hoc networks*, Computer Communications **23** (1999), 1627–1637.

[5] P. Bahl, R. Chandra, and J. Dunagan, *Ssch: slotted seeded channel hopping for capacity improvement in ieee 802.11 ad-hoc wireless networks*, Proceedings of the 10th annual international conference on Mobile computing and networking (New York, NY, USA), MobiCom '04, ACM, 2004, pp. 216–230.

[6] R. Bellman, *On a Routing Problem*, Quarterly of Applied Mathematics **16** (1958), 87–90.

[7] F.C. Berry, B.A. Black, P.S. DiPiazza, B.A. Ferguson, and D.R. Voltmer, *Introduction to wireless systems*, 1st ed., Pearson Education, 2008.

[8] L. Bononi, M.D. Felice, A. Molinaro, and S. Pizzi, *Joint channel assignment and multi-path routing for multi-radio wireless mesh networks*, Proceedings of the 2009 29th IEEE International Conference on Distributed Computing Systems Workshops (Washington, DC, USA), ICDCSW '09, IEEE Computer Society, 2009, pp. 476–481.

[9] M. Burkhart, P. von Rickenbach, R. Wattenhofer, and A. Zollinger, *Does topology control reduce interference?*, Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing (New York, NY, USA), MobiHoc '04, ACM, 2004, pp. 9–19.

[10] S. Chan, R. Poovendran, and M. Sun, *A key management scheme in distributed sensor networks using attack probabilities*, Global Telecommunications Conference, 2005. GLOBECOM '05. IEEE, 2005.

[11] M. Chen, V. Leung, C.M. Victor, S. Mao, and Y. Yuan, *Directional geographical routing for real-time video communications in wireless sensor networks*, Comput. Commun. **30** (2007), 3368–3383.

[12] M. Chen, V.C.M. Leung, S. Mao, and Y. Yuan, *Directional geographical routing for real-time video communications in wireless sensor networks*, Comput. Commun. **30** (2007), 3368–3383.

[13] S.L. Chen, P.H. Chong, and M. Yang, *Dynamic channel assignment with flexible reuse partitioning in cellular systems*, Wirel. Pers. Commun. **42** (2007), 161–183.

[14] S. Cho and C. Kim, *Interference-aware multi-channel assignment in multi-radio wireless mesh networks.*, IEICE Transactions.

[15] J.D. Deaton, S.A. Ahmad, U. Shukla, R.E. Irwin, L.A. Dasilva, and A.B. Mackenzie, *Evaluation of dynamic channel and power assignment for cognitive networks*, Wirel. Pers. Commun. **57** (2011), 5–18.

[16] E.W. Dijkstra, *A note on two problems in connexion with graphs*, Numerische Mathematik **1** (1959), 269–271, 10.1007/BF01386390.

[17] Y. Ding, Y. Huang, G. Zeng, and L. Xiao, *Channel assignment with partially overlapping channels in wireless mesh networks*, Proceedings of the 4th Annual International Conference on Wireless Internet (ICST, Brussels, Belgium, Belgium), WICON '08, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, pp. 38:1–38:9.

[18] R. Draves, J. Padhye, and B. Zill, *Routing in multi-radio, multi-hop wireless mesh networks*, Proceedings of the 10th annual international conference on Mobile computing and networking (New York, NY, USA), MobiCom '04, ACM, 2004, pp. 114–128.

[19] X. Du, Y. Xiao, M. Guizani, and H. Chen, *An effective key management scheme for heterogeneous sensor networks*, Ad Hoc Networks **5** (2007), no. 1, 24 – 34, Security Issues in Sensor and Ad Hoc Networks.

[20] P. Ebinger and M. Parsons, *Measuring the impact of attacks on the performance of mobile ad hoc networks*, Proceedings of the 6th ACM symposium on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks (New York, NY, USA), PE-WASUN '09, ACM, 2009, pp. 163–164.

[21] L. Eschenauer and V.D. Gligor, *A key-management scheme for distributed sensor networks*, Proceedings of the 9th ACM conference on Computer and communications security (New York, NY, USA), CCS '02, ACM, 2002, pp. 41–47.

[22] R. Flickenger, *Wireless networking in the developing world*, 2008.

[23] L. Gao, E. Chang, S. Parvin, S. Han, and T. Dillon, *A secure key management model for wireless mesh networks*, Proceedings of the 2010 24th IEEE International Conference on Advanced Information Networking and Applications (Washington, DC, USA), AINA '10, IEEE Computer Society, 2010, pp. 655–660.

[24] M.R. Garey and D.S. Johnson, *Computers and intractability; a guide to the theory of np-completeness*, W. H. Freeman & Co., New York, NY, USA, 1990.

[25] Software GmbH, *Leda: Algorithmic solutions*, December 2011.

[26] P. Gupta and P.R. Kumar, *The capacity of wireless networks*, Information Theory, IEEE Transactions on **46** (2000), no. 2, 388 –404.

[27] A.H. Hać and C.H. Mo, *Dynamic channel assignment in wireless communication networks*, Int. J. Netw. Manag. **9** (1999), 44–66.

[28] X. Hu and M.J. Lee, *An efficient multipath structure for concurrent data transport in wireless mesh networks*, Comput. Commun. **30** (2007), 3358–3367.

[29] P. Huang, H. Tian, M. Zhang, and P. Zhang, *Robust multi-path routing for dynamic topology in wireless sensor networks*, The Journal of China Universities of Posts and Telecommunications **14** (2007), no. 1, 1 – 5.

[30] IEEE, *Ieee standard for information technology–telecommunications and information exchange between systems–local and metropolitan area networks–specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments*, IEEE Std 802.11p-2010 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, IEEE Std 802.11n-2009, and IEEE Std 802.11w-2009) (2010), 1 –51.

[31] K. Jain, J. Padhye, V.N. Padmanabhan, and L. Qiu, *Impact of interference on multi-hop wireless network performance*, Proceedings of the 9th annual international conference on Mobile computing and networking (New York, NY, USA), MobiCom '03, ACM, 2003, pp. 66–80.

[32] F. Kandah, W. Zhang, X. Du, and Y. Singh, *A secure key management scheme in wireless mesh networks*, Communications (ICC), 2011 IEEE International Conference on, june 2011, pp. 1 –5.

[33] F. Kandah, W. Zhang, Y. Singh, and J. Li, *Interference-aware robust wireless mesh network design*, GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference, dec. 2010, pp. 1 –5.

[34] F. Kandah, W. Zhang, C. Wang, and J. Li, *Diverse path routing with interference and reusability consideration in wireless mesh networks*, Mobile Networks and Applications, 1–10, 10.1007/s11036-011-0301-y.

[35] P. Kyasanur and N.H. Vaidya, *Routing and interface assignment in multi-channel multi-interface wireless networks*, Wireless Communications and Networking Conference, 2005 IEEE, vol. 4, 2005, pp. 2051 – 2056 Vol. 4.

[36] W.H. Lehr and J.M. Chapin, *On the convergence of wired and wireless access network architectures*, Information Economics and Policy **22** (2010), no. 1, 33–41.

[37] J. Liu, X. Jiang, H. Nishiyama, and N. Kato, *Reliability assessment for wireless mesh networks under probabilistic region failure model*, Vehicular Technology, IEEE Transactions on **PP** (2011), no. 99, 1.

[38] P. Loree, K. Nygard, and X. Du, *An efficient post-deployment key establishment scheme for heterogeneous sensor networks*, Proceedings of the 28th IEEE conference on Global telecommunications (Piscataway, NJ, USA), GLOBECOM'09, IEEE Press, 2009, pp. 5370–5375.

[39] C. Maple, G. Williams, and Y. Yue, *Reliability, availability and security of wireless networks in the community*, 2007.

[40] M. Médard, S.G. Finn, and R.A. Barry, *Redundant trees for preplanned recovery in arbitrary vertex-redundant or edge-redundant graphs*, IEEE/ACM Trans. Netw. **7** (1999), 641–652.

[41] A. Mishra, S. Banerjee, and W. Arbaugh, *Weighted coloring based channel assignment for wlans*, SIGMOBILE Mob. Comput. Commun. Rev. **9** (2005), no. 3, 19–31.

[42] K. Moaveni-nejad and X. Li, *Low-interference topology control for wireless ad hoc networks*, ACM Wireless Networks, IEEE Press, 2005.

[43] A.B. Mohanoor, S. Radhakrishnan, and V. Sarangan, *Interference aware multi-path routing in wireless networks*, Mobile Ad Hoc and Sensor Systems, 2008. MASS 2008. 5th IEEE International Conference on, 29 2008.

[44] A. Morgado, R. del Río, , and J.M. de la Rosa, *A triple-mode reconfigurable sigma-delta modulator for multi-standard wireless applications*, Proceedings of the conference on Design, automation and test in Europe (New York, NY, USA), DATE '08, ACM, 2008, pp. 862–867.

[45] H. Moustafa, U. Javaid, T. Rasheed, and D. Meddour, *A panorama on wireless mesh networks: Architectures, applications and technical challenges*, 2006.

[46] N.S. Nandiraju, D.S. Nandiraju, and D.P. Agrawal, *Multipath routing in wireless mesh networks*, Mobile Adhoc and Sensor Systems (MASS), 2006 IEEE International Conference on, 2006, pp. 741 –746.

[47] L. Narayanan, *Handbook of wireless networks and mobile computing*, John Wiley & Sons, Inc., New York, NY, USA, 2002, pp. 71–94.

[48] S. Nelakuditi and Z. Zhang, *On selection of paths for multipath routing*, Proceedings of the 9th International Workshop on Quality of Service (London, UK), IWQoS '01, Springer-Verlag, 2001, pp. 170–186.

[49] A.H.M. Rad and V.W.S. Wong, *Joint channel allocation, interface assignment and mac design for multi-channel wireless mesh networks*, INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE, May 2007, pp. 1469 –1477.

[50] A. Raniwala and T. Chiueh, *Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network*, INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, vol. 3, 2005, pp. 2223 – 2234 vol. 3.

[51] A. Raniwala, K. Gopalan, and T. Chiueh, *Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks*, SIGMOBILE Mob. Comput. Commun. Rev. **8** (2004), 50–65.

[52] T. Schmid, T. Dreier, and M.B. Srivastava, *Software radio implementation of short-range wireless standards for sensor networking*, Proceedings of the 4th international conference on Embedded networked sensor systems (New York, NY, USA), SenSys '06, ACM, 2006, pp. 381–382.

[53] I. Sheriff and E. Belding-Royer, *Multipath selection in multi-radio mesh networks*, Broadband Communications, Networks and Systems, 2006. BROADNETS 2006. 3rd International Conference on, 2006, pp. 1 –11.

[54] J. Shi, R. Zhang, and Y. Zhang, *Secure range queries in tiered sensor networks.*, INFOCOM'09, 2009, pp. 945–953.

[55] D.M. Shila and T. Anjali, *Defending selective forwarding attacks in wmns*, Electro/Information Technology, 2008. EIT 2008. IEEE International Conference on, may 2008, pp. 96 –101.

[56] J. So and N. Vaidya, *A routing protocol for utilizing multiple channels in multi-hop wireless networks with a single transceiver*, UIUC Technical Report.

[57] A. Srinivas and E. Modiano, *Minimum energy disjoint path routing in wireless ad-hoc networks*, in Proceedings of the 9th Annual International Conference on Mobile Computing and Networking, 2003, pp. 122–133.

[58] A.P. Subramanian, H. Gupta, and S.R. Das, *Minimum interference channel assignment in multi-radio wireless mesh networks*, Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON '07. 4th Annual IEEE Communications Society Conference on, 2007, pp. 481 –490.

[59] W. Sun, Z. Qin, L. Yao, and M. Li, *Research on interference-based channel assignment methods in 802.11-based wireless mesh network*, Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on, 2008, pp. 1 –4.

[60] H. Tan, T. Lou, F.C.M Lau, Y. Wang, , and S. Chen, *Minimizing interference for the highway model in wireless ad-hoc and sensor networks*, Proceedings of the 37th international conference on Current trends in theory and practice of computer science (Berlin, Heidelberg), SOFSEM'11, Springer-Verlag, 2011, pp. 520–532.

[61] J. Tang, G. Xue, and W. Zhang, *Interference-aware topology control and qos routing in multi-channel wireless mesh networks*, Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing (New York, NY, USA), MobiHoc '05, ACM, 2005, pp. 68–77.

[62] J.W. Tsai and T. Moors, *Interference-aware multipath selection for reliable routing in wireless mesh networks*, Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on, 2007, pp. 1 –6.

[63] M. Čagalj, J. Hubaux, and C. Enz, *Minimum-energy broadcast in all-wireless networks: Np-completeness and distribution issues*, Proceedings of the 8th annual international conference on Mobile computing and networking (New York, NY, USA), MobiCom '02, ACM, 2002, pp. 172–182.

[64] _____, *Minimum-energy broadcast in all-wireless networks: Np-completeness and distribution issues*, Proceedings of the 8th annual international conference on Mobile computing and networking (New York, NY, USA), MobiCom '02, ACM, 2002, pp. 172–182.

[65] S. Waharte, B. Ishibashi, R. Boutaba, and D. Meddour, *Design and performance evaluation of iar: Interference-aware routing metric for wireless mesh networks*, Mob. Netw. Appl. **14** (2009), 649–660.

[66] C. Wu, *Hybrid dynamic channel assignment in clustered wireless multihop cdma/tdma ad hoc networks*, Wirel. Pers. Commun. **42** (2007), 85–105.

[67] C. Wu, F. Zhang, and H. Yang, *A novel qos multipath path routing in manet.*, JDCTA.

[68] H. Xia and J.C. Brustoloni, *Hardening web browsers against man-in-the-middle and eavesdropping attacks*, Proceedings of the 14th international conference on World Wide Web (New York, NY, USA), WWW '05, ACM, 2005, pp. 489–498.

[69] S. Xiao, W. Gong, and D. Towsley, *Secure wireless communication with dynamic secrets*, Proceedings of the 29th conference on Information communications (Piscataway, NJ, USA), INFOCOM'10, IEEE Press, 2010, pp. 1568–1576.

[70] W. Zhang, F. Kandah, J. Tang, and K. Nygard, *Interference-aware robust topology design in multi-channel wireless mesh networks*, Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE, jan. 2010, pp. 1 –5.

[71] W. Zhang, G. Xue, J. Tang, and K. Thulasiraman, *Faster algorithms for construction of recovery trees enhancing qop and qos*, Networking, IEEE/ACM Transactions on **16** (2008), no. 3, 642 –655.

[72] X. Zhao, Y. Lv, T.H. Yeap, and B. Hou, *A novel authentication and key agreement scheme for wireless mesh networks*, Proceedings of the 2009 Fifth International Joint Conference on INC, IMS and IDC (Washington, DC, USA), NCM '09, IEEE Computer Society, 2009, pp. 471–474.