A Study on Distributed Management Schemes for Node Mobility in the Internet

A dissertation submitted to
the Graduate School of Science and Technology of Keio University, Japan
in partial fulfillment of the requirements for the degree of
Ph.D in Engineering
September, 2015

HenakaRalalage Oshani Erunika

Acknowledgment

This piece of writing would have never been a reality without unlimited love and support of many special figures. I would like to take this chance to acknowledge them for the faith that they had on me and the immense support they have offered me.

Most specially, I extend my heartfelt gratitude to my parents, husband, and my brother, who were with me through thick and thin, showing me unconditional love and support. They have been my strength, motivation, and courage, for which I will never be able to thank them enough. I should specially mention the encouragement and assistance I have been receiving from my husband and his patience, which allowed me to realize my dream.

Keio University has been my home for the last three years from 2012 to 2015. From the very first day I arrived in Japan, It was never a smooth ride for me in many different ways. Everything seemed to be new and challenging. My supervisor, Prof. Fumio Teraoka should be given much credit for the warm welcoming helping hand he extended through out my research carrier at Keio University. Further, not forgetting Prof. Kunitake Kaneko for the support that he extended during the lab life. I appreciate their patience and kind advices which made my research a reality. Comments, recommendation, and thoughts that I received from the Ph.D committee should specially be mentioned with extreme gratitude. Thus, I would like to extend my acknowledgment to Prof. Hiroshi Shigeno and Prof. Takahiro Yakoh. The ultimate outcome of my research would not have been put into words like this, in such a successful manner without their support. Also, my lab mates played a huge role in my success with their never ending support. I would like to express my gratefulness to all the Teraoka-Kaneko laboratory members (2012-2015).

I cannot forget the backup and support I received back at home, at University of Peradeniya, Sri Lanka. Prof. Saluka Kodituwakku, Prof. Pushpa Wijekoon, and Prof. Thewarapperuma have been role models of my life ever since I entered the Department of Statistics and Computer Science. Without their guidance, I would not have walked this path. Further, I would like to remind the wholehearted guidance and support I received from Dr. Amalka Pinidiyaarachchi and Dr. Roshan Yapa, not forgetting Dr. Giritheran Balathasan, Dr. Ruwan Nawaratna Dr. Mohomad Fazeen, and Dr. Jayantha Kumara.

Last, but certainly not least, there has been an innumerable amount of people involved in this endeavor, whom I have failed to mention by names. With a heartfelt apology, I wish to extend my acknowledgment for them.

Abstract

This thesis introduces a comprehensive study carried out focusing on Mobility Management (MM) schemes in the Internet. It describes the evaluation procedure adopted along with the simulator which is constructed to support the evaluation. Then, the results obtained are explained, which lead to introduce a novel MM scheme.

During the past couple of decades, there has been a considerable number of standardization efforts for MM in the Internet. However, those standard methods severely suffer from well-known problems. Those problems include a single point of failure and attack, non-optimal data routing, and restricted scalability. Even though there has been a significant amount of efforts which try to identify such major problems and fix them by adopting a Distributed MM (DMM) approach, those efforts are still in the proposal phase.

Performance evaluation of MM protocols remains challenging, despite of having a handful of network simulators. These simulators provide least conveniences to build and evaluate new protocols, regardless of having a good set of facilities to simulate ordinary standard protocols. Especially, it is difficult to simulate relatively new concepts, such as DMM. Thus, it remains extremely strenuous to evaluate proposed schemes and identify their competencies and applicability.

Motivated by the lack of performance evaluation efforts and lack of simulation support, an Internet Protocol (IP) mobility simulator called SimNetDMM was designed considering the network layer of the Internet. It attempts to simulate mobility scenarios over mapped and synthetic topologies for all the selected MM schemes. SimNetDMM can be regarded as the inceptive effort of that sort. It also allows realistic mobility patterns, MM entity installments, and routing policies. Thus, the results can be considered closely liable and realistic.

The selected set of client driven (host-based) and network driven (network-based) MM schemes are evaluated for performance separately. Evaluation carried out for host-based MM schemes reveals that the fully distribution in the control-plane in terms of functionality retrieves better control-plane performance. On the other hand, better performance in data-plane is observed during the evaluation for network-based DMM schemes with control/data-plane split. Thus, in overall, fully functional distribution in the control plane and control/data-plane split are identified as candidate MM approaches. Disadvantages of the existing proposals focusing on these concepts are also identified. A fully distributed host-based MM scheme found in the literature lacks consistency. The scheme adopting control/data-plane split has poor control-plane performance and when employed in a distributed control-plane environment, it suffers from heavy control overhead due to data residency.

Considering the above issues, a novel localized network-based fully-distributed MM scheme (DMMSDN) is introduced. It is designated for a distributed Software Defined Network (SDN) environment. Control/data-plane separation is achieved with SDN. Dynamic Host Configuration Protocol for IPv6 (DHCPv6) is adopted for accounting and assigning IPv6 addresses for Mobile Nodes (MNs). DMMSDN distributes the control-plane reducing data redundancy and increasing consistency of MM. Further, it regulates flow table updates in SDN to reduce control-plane overhead. The size of memory required in SDN-controllers is minimized by limiting MN's mobility information to be stored only in a single SDN-controller called the initial SDN-controller.

DMMSDN is simulated and evaluated against a few DMM schemes. The results confirm the applicability of DMMSDN. Further, multiple SDN-controller installment is also examined. Optimal SDN-controller installments are identified for different topologies. Highest distribution in the control-plane is better for larger Internet Service Provider (ISP) networks residing closer to the Internet core (tier-1), which cover a few continents and have a considerably large number of Access Routers (ARs) for MNs. Medium distribution is better for least distributed ISP topologies residing at the edge of the Internet (tier-3). Least distribution is better for medium ISP topologies (tier-2). They are tend to lease Internet Access to edge ISPs rather than providing direct access to MNs. The results confirm that multiple SDN-controller installment always outperforms single SDN-controller installment. Further, control-plane results are stable despite of the inter-communication of SDN-controller set. Thus, it assures the admissibility of DMMSDN.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Mobility Management (MM) in the Internet has become a mandatory networking aspect on the grounds that a rapid growth is observed in the usage of portable devices to access the Internet. Thus, it can be considered as one of the most demanding and challenging aspects as far as today's Internet Service Providers (ISPs) are concerned. Considering the inadequacy of the standard centralized MM approaches for the reason that they suffer a single point of failure, non-optimal routing and scalability limitations, Distributed MM (DMM) was introduced. However, being a comparatively novel notion, DMM is yet to establish standardization. Most DMM schemes that are found in the literature are drafts or proposals which lack implementation, prototyping, and testing. Moreover, the literature lacks studies based on these proposed DMM schemes in terms of performance evaluation and comparison. One of the main reasons for this is perceived to be the lack of evaluation support. Modern DMM concepts are least facilitated by generic simulators, making the evaluation of proposed schemes even complicated. This thesis focuses mainly on evaluating DMM schemes found in the literature envisioning to determine the best approach. Then, it introduces a novel DMM scheme called DMMSDN, which tries to overcome the disadvantages found in the already proposed schemes.

In order to obtain a better comprehension of the matters discussed in this thesis, it is extremely important to identify the background of the work that is conveyed in succeeding chapters. Thus, the rest of this chapter is organized to give a brief insight to the background of this study. Then, the motivation of this study and the contributions are introduced.

1.1 Background of the Mobility Management Environment

1.1.1 Internet Access and User Mobility

Looking back in time, it might amaze someone how rapid the Internet has grown and how it has blended closely with the day-to-day life. Along the past couple of decades, the Internet has evolved to make everyday life convenient by virtualizing regular activities as Internet-based services. Today, a common user would use the Internet for many purposes, starting from sending an e-mail to something much complex such as a financial transaction.

It is worth exploring what might have complemented this rapid popularity of the Internet. Clearly, it is the the broad accessibility and accommodating nature of the technologies around which the Internet evolved. On one hand, the networking technologies have grown beyond imaginary boundaries. Starting from a very basic layout allowing a remote terminal to communicate with the Central Processing Unit (CPU), networking has grown rapidly within just a couple of decades to allow the communication between any terminal, lying any distance apart, with higher bandwidths. On the other hand, devices used to access the Internet also have undergone a drastic reformation. A couple of decades earlier, it was a stationary desktop computer which was used commonly to access the Internet. But today, it is not difficult to find a device which can even fit in someone's pocket, that is capable of providing the Internet access, anytime, anywhere. This makes Internet access simpler, quicker, and cost effective.

As a consequence of using multiple devices such as mobile phones, smart phones, laptops, and tablet Personal Computers (PCs), has urged the necessity to access common data through different devices. This has introduced the concept of network storage. It allows all the data of the user to be stored in the Internet rather than in a specific device. Then, data can be accessed by any device, just by connecting to the Internet. Thus, the required storage space in portable devices has shrunk. It has allowed vendors to improve the other aspects, while minimizing the storage. As a result, more sophisticated portable Internet devices have been introduced. It can be seen as one of the most assistive factors that boosts the Internet access of individuals. With the convenience the portable devices provide, Internet access via portable devices might totally occupy the statistics in the near future.

Looking at this scenario from the network perspective, it urges the necessity to provide a specific device (node) with reachability, despite of its physical movement. This movement may cause the node to change its point of Internet access frequently. Thus, the node is likely to under go detachments and re-attachments from different Internet Access Points (APs), known as handovers. Uninterrupted communication provision (session continuity) during an encounter of a handover is crucial in maintaining user experience. The next section, Sect. 1.1.2 looks at the Internet data routing to understand the challenges that arise when attempting to facilitate mobility.

1.1.2 Data Routing in the Internet

When the physical composition of the Internet is considered, it can be seen as a globally interconnected computer network system. In simple terms, it is a network of networks combined in a hierarchical manner. To reduce the complexity of the software architectural design, it is organized in a stacked of layers. Each layer is performing a specific task towards networking. A layer can be composed of single to multiple software components known as protocol entities. The layer of the network architecture which deals with delivering data packets from the source node to destination node is recognized as the *network layer* (L3). Figure 1.1 depicts a simple representation of the Internet from the perspective of the network-layer.

In the network-layer, the Internet can be seen as a composition of Internet Service Provider (ISP) networks. From the administration perspective, Autonomous System

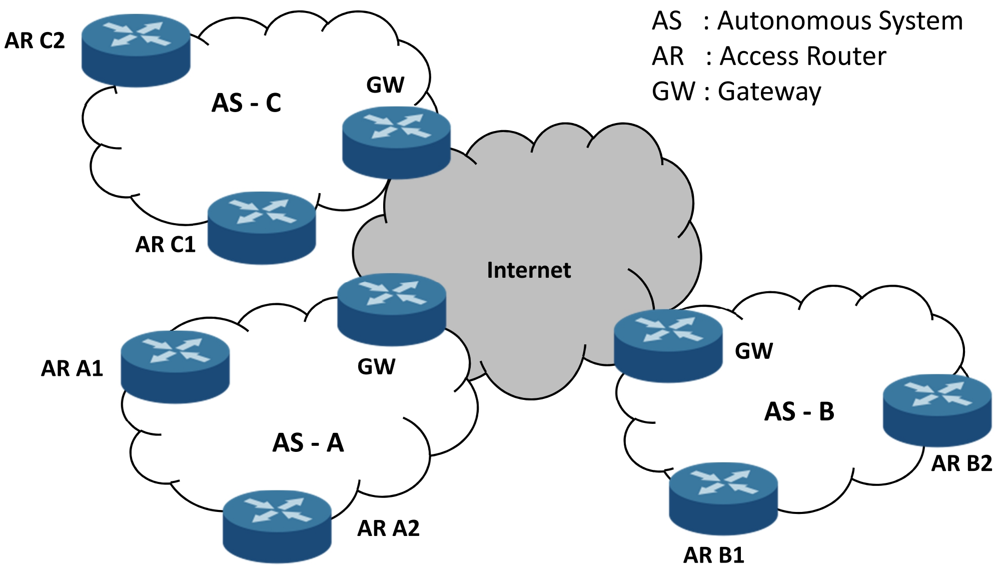


Figure 1.1: Brief Overview of the Internet According to the Network Layer.

(AS) represents a single administrative portion of the Internet which exposes itself as a clearly defined unit. Thus, an ISP can be composed of either single or multiple ASes. AS-A, AS-B, and AS-C represent three distinct ASes in Fig. 1.1. ASes are connected to other Packet Data Network (PDN) (the Internet according to Fig. 1.1) via the Gateway (GW) which is a peering point. An AS can have multiple GWs.

The main component which holds the entire Internet together can be considered as the common protocol employed in the network layer. It is known as the Internet Protocol (IP). By using a uniform protocol like IP, it has been able to bring together a massive number of heterogeneous devices scattered around the world. This standard protocol specifies the packet format and the node identification techniques used to allow the node communication over the network. In the current context, two IP versions are widely used. Being the former standardization, IP version 4 (IPv4) [52] is technically capable of identifying 2^{32} devices with a 32-bit addressing scheme. However, the number of devices IPv4 actually addresses is restricted due to different administrative constraints. It uses the packet or PDU (Protocol Data Unit) called IPv4 packets. Even though IPv4 was sustainable during early ages of the Internet, due to the rapid growth of the Internet, IPv4 address space was anticipated to exhaust. To solve IPv4 address exhaustion, IP version 6 (IPv6) [29] was introduced by the Internet Engineering Task Force (IETF) [30]. It uses a 128-bit addressing scheme which is capable of addressing 2^{128} devices. Similar to the IPv4, IPv6 is also restricted in terms of the actual number of devices it can address. The PDU format used in IPv6 is known as IPv6 packets.

An end host node accesses to the ISP network using an Access Router (AR) belonging to the AS. Then the node is assigned an IPv4 / IPv6 address using the addressing schemes depicted in Figs. 1.2 and 1.4 in IPv4 and IPv6, respectively. This addressing mechanism enables to locate the node within the Internet by allowing the aggregation

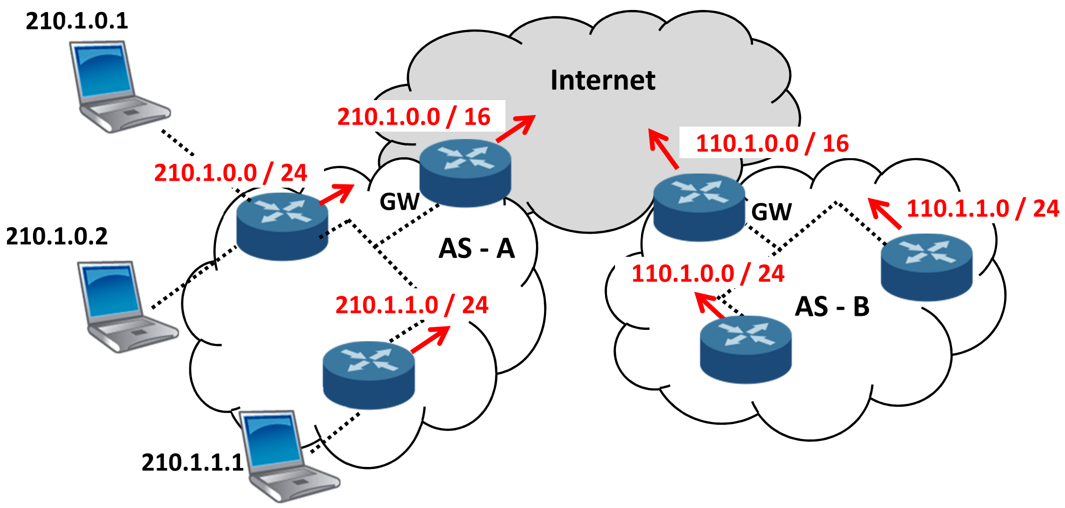


Figure 1.2: IPv4 Address Assignment.

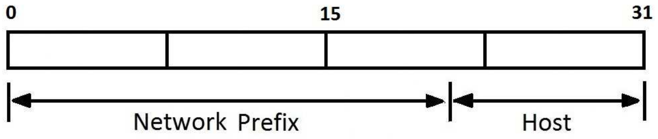


Figure 1.3: IPv4 Addressing Scheme.

of IPv4 / IPv6 addresses. Figure 1.3 shows an example global IPv4 address allocation, whereas Fig. 1.5 shows an instance of global IPv6 address allocation. Each AS holds a block of IPv4/IPv6 addresses which it advertises to the rest of the Internet using the *network prefix*. Network prefix is the common part of the IPv4 / IPv6 addresses shared by all the nodes belonging to that AS. Similarly, if the AS is composed of subnetworks, each subnetnetwork will have a specific block of IPv4/IPv6 addresses which it advertises towards the core of the AS.

During packet exchange, the IPv4 / IPv6 address serves as the identity and the locator of the node at the same time. When packets enter the Internet, it is vital to identify and locate the receiver node in order to deliver the packet, correctly. In other terms, the packet delivering mechanism should be able to resolve the identity of the receiving device first. Then, it should determine how to reach that specific node via the complex Internet. By using the IPv4 / IPv6 address scheme, this task has been simplified. An example packet delivery is shown in Fig. 1.6.

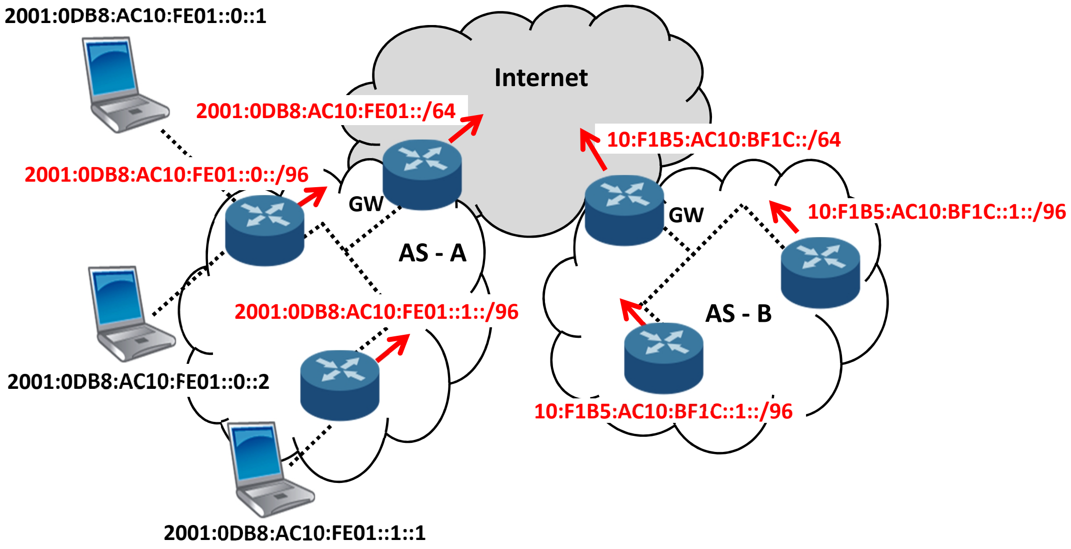

2001:0DB8:AC10:FE01::0::1
Internet
2001:0DB8:AC10:FE01::/64
10:F1B5:AC10:BF1C::/64
2001:0DB8:AC10:FE01::0::/96
GW
GW
10:F1B5:AC10:BF1C::1::/96
AS - A
AS - B
2001:0DB8:AC10:FE01::1::/96
2001:0DB8:AC10:FE01::0::2
10:F1B5:AC10:BF1C::1::/96
2001:0DB8:AC10:FE01::1::1

Figure 1.4: IPv6 Address Assignment.

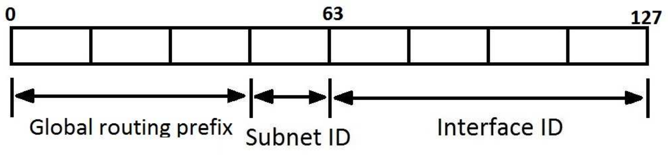


Figure 1.5: IPv6 Addressing Scheme.

1.2 Mobility Management in the Internet

1.2.1 Requirement for Mobility Management in the Internet

Current model of packet delivery adopted in the Internet uses the IPv4 / IPv6 address for two purposes. It serves as the identifier and the locator of the node at the same time. If the node changes its attachment location to the Internet, it is assigned a new IPv4 / IPv6 address based on the new location. This leads to automatically loose the identity it bears at the previous location. As a consequence, all the communication sessions the Mobile Node (MN) had with other Correspondent Nodes (CNs) using the previous IPv4 / IPv6 address may terminate. Moreover, the MN becomes unreachable through the previously assigned IPv4 / IPv6 address. This scenario is depicted in Fig. 1.7.

With the popularity of portable devices, this scenario has become a common encounter. Thus, a mechanism was immensely required in order to provide session continuity and reachability for such Mobile Nodes (MNs). The basic requirements for mobility management can be listed as follows.

1. MNs should be able to use the IPv4 / IPv6 address it is assigned anywhere, re-

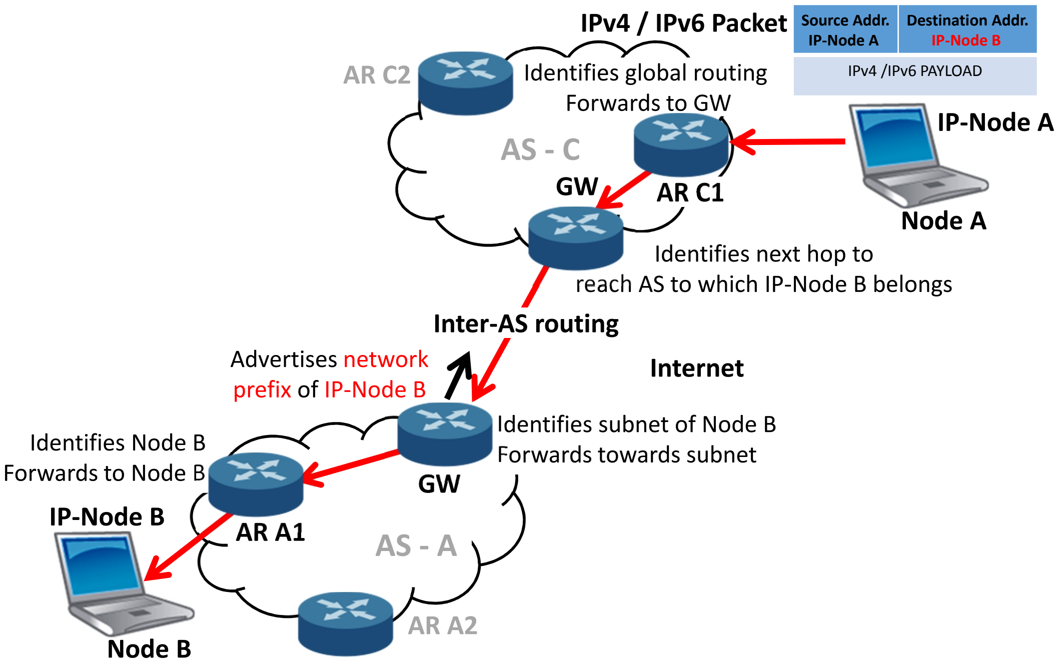


Figure 1.6: Packet Delivery in the Internet.

gardless of its attachment location to the Internet.

2. The schemes should not make any modifications for non-mobile nodes.

3. Mobility management should not modify underlying layers.

4. Packets that are destined to the MN should make least detours during its delivery.

5. Solution should be scalable.

6. No overhead should incur due to mobility management when the MN is in its original location specified by the IPv4 / IPv6 address it bares.

7. Mobility management should be consistent.

1.2.2 Mobility Management in the Internet

Mobility Management addresses the problem arising when trying to use an IPv4 / IPv6 address to identify, as well as to locate a specific node and its sessions. The main idea is to offer transparency to the movement of the MN, such that session continuity and reachability are granted despite of the MN's consequent locations of attachment (Fig.

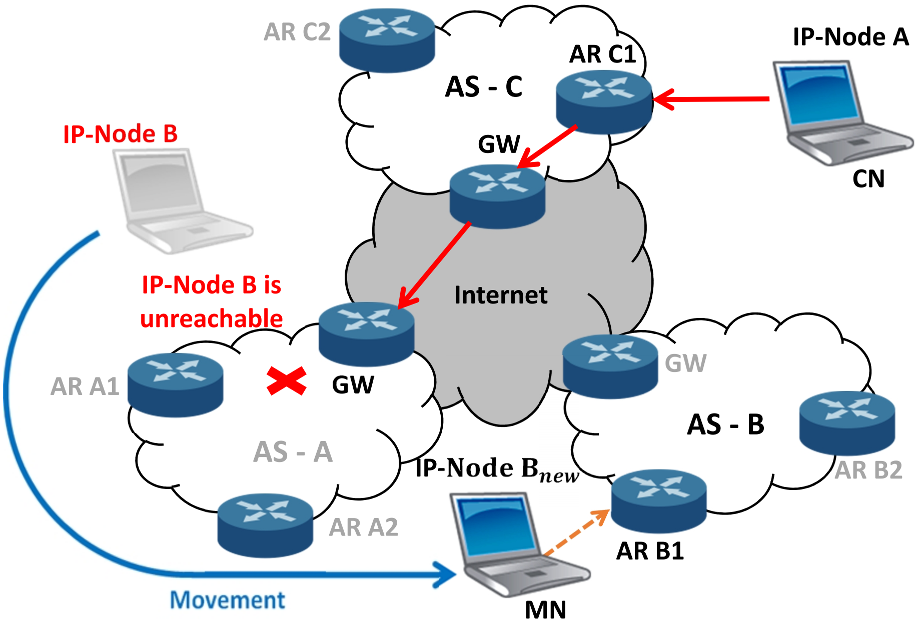


Figure 1.7: Node Mobility in the Internet.

1.8). There can be seen a wide variety of attempts that tries to meet the requirements of mobility management in the literature. Mobile IPv4 (MIPv4) [50], Network Mobility (NEMO) [37], MIPv6 [51], Hierarchical MIPv6 (HMIPv6)[61], Proxy MIPv6 (PMIPv6) [25] can be considered as standardized protocols that are commonly encountered. They all try to separate the identity it uses for session continuity despite of relocation from its locator. Further, a centralized entity is introduced to keep tracks of the MN's movement (binding) and to redirect the incoming packets destined to the IPv4 / IPv6 address of the MN while MN is away from the home network.

An example scenario where session continuity and reachability of the MN is handled using MIPv6 is given in Fig. 1.9. Further details are given in Sect. 2.3. It illustrates an instance of the Correspondent Node *CN* communicating with the Mobile Node *MN*, which resides at the *home network*. It is assigned an IPv6 address called the *Home Address* (HoA), which is mobility enabled. During the session, *MN* moves to a different AS. Then, it attains a new IPv6 address called the *Care of Address* (CoA). This is informed to the *Home Agent* (HA), which resides in the *home network* of the MN. It stores the relationship between then HoA and the CoA as a *binding*.

When packets destined to HoA arrives at the *home network*, the HA intercepts and tunnels them to the *MN*, which is now reachable at the CoA.

However, in spite of being commonly used, the above mentioned standardized protocols have well known problems. Figure 1.9 demonstrates some of the problems encountered by conventional MM schemes. Non-optimal packet routing between CN and MN is one of the major problems in this architecture. This is due to the packet detour that occur since of the packet interception of the HA. Then, a single point of failure due

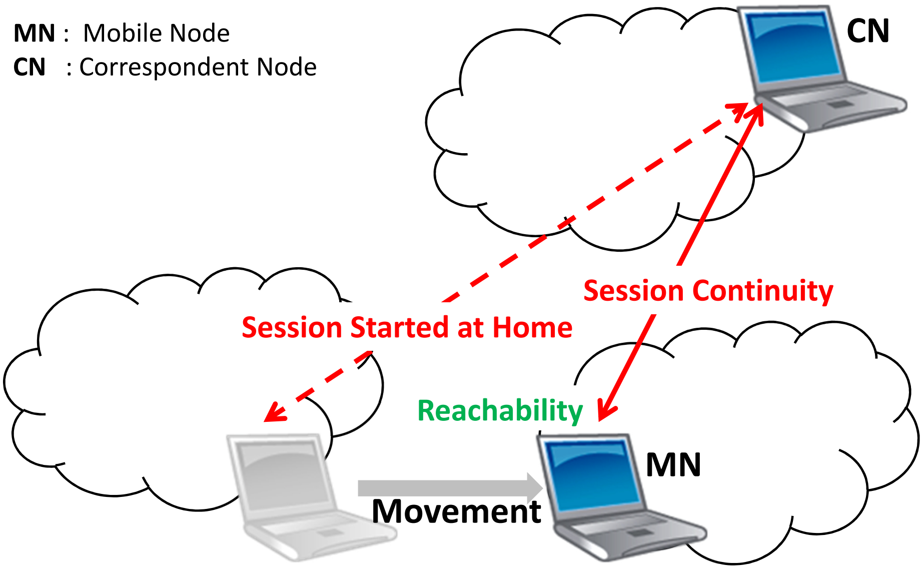


Figure 1.8: Mobility Management in the Internet.

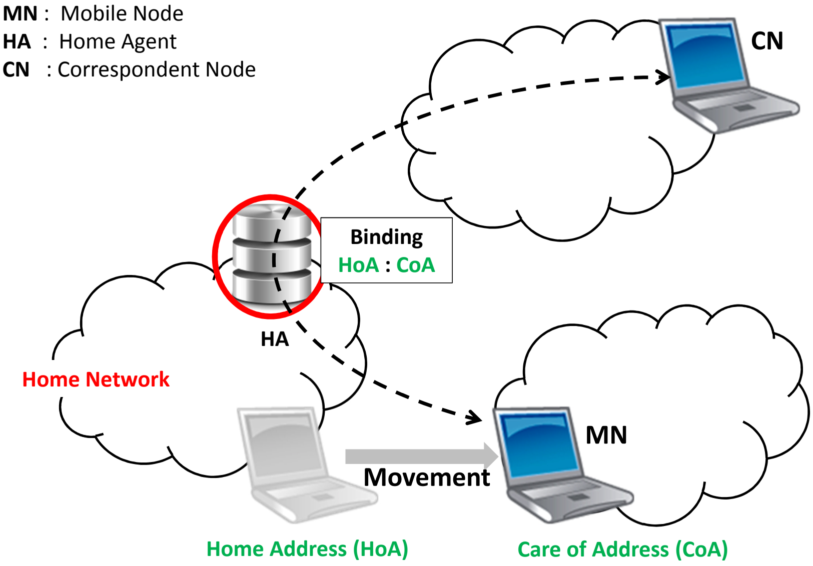


Figure 1.9: Standard Mobility Management Approach (Mobile IPv6).

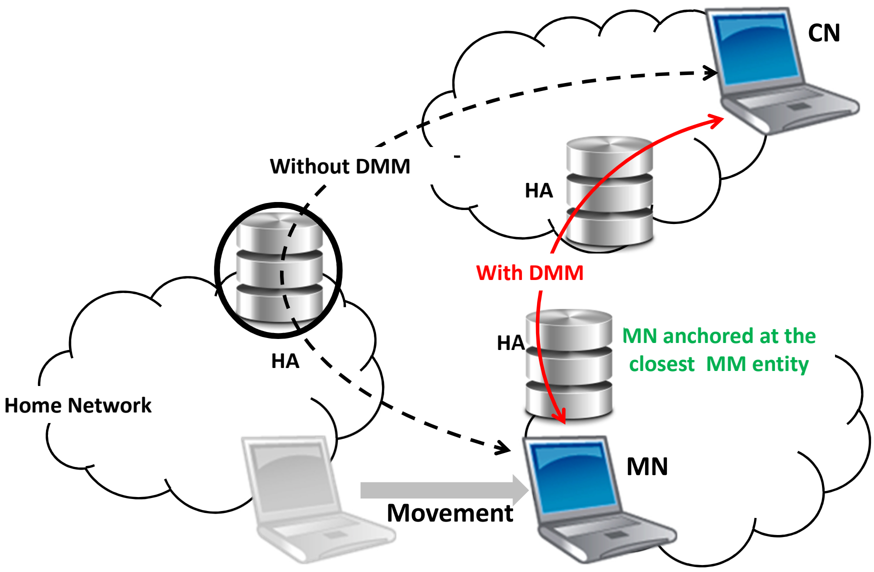


Figure 1.10: Distributed Mobility Management Approach (Global HAHA/Migrating HA).

to having a centralized agent handling mobility (Fig. 1.9 - HA) is also a critical problem. Further, when the demand for mobility increases, subscribers per HA increases. Because of the limited amount of resources available in the centralized agent facilitating mobility, scalability will be restricted. There have been plenty of attempts to address these problems in the literature recently. Majorly, they focus on distributing the functionality of MM in the Internet, known as Distributed Mobility Management (DMM). This concept is relatively young. Figure 1.10 shows a typical scenario of DMM, where MM entity is duplicated in the Internet.

The Internet Engineering Task Force (IEFT) [30] nurtures a working group dedicated to form and shape DMM, know as the DMM Working Group (DMM-WG) [21]. The DMM-WG has made a tremendous effort to bring together the scattered ideas and build a frame work for DMM. One of the main targets of the DMM-WG is to establish standards for DMM, which is yet to be completed. It has already published two Request for Comments (RFC) documents. One of them defines requirements for DMM [59], whereas the other gives an account of current practices and provides a gap analysis [38]. Currently, the DMM-WG does not foster any proposed scheme as a working group document. Thus, the standardization of DMM schemes can be seen to lag behind. Nevertheless, it has facilitated a considerable amount of related Internet drafts, creating an open discussion ground on DMM. Such discussions include novel MM ideas inclusing treating the control related and data packets separately (control/data-plane split) or breaking down the mobility handling procedure into a set of functions and distributing them (functional distribution). The mentioned Internet drafts are still in the proposal state, where prototyping, testing, and implementation are yet to be performed.

Thus, it remains difficult to determine applicability and appropriateness of the proposed schemes.

1.3 Motivation of this Study

In computer networking, it is vital to examine the behavior of networks in different conditions to identify the effectiveness of a deployed protocol. Simulation can be considered as one of the major methods through which it is achieved. Network simulation models the underlying states of the network to predict the behavior using mathematical models. The Network Simulator 3 (NS-3) [46], NetSim [44], and OPNET [48] can be considered as well known simulators available for network simulation. Even though, they are capable of simulating mobility protocols in the standard space, they provide least support to simulate non-standard mobility protocols. More to the point, it is extremely challenging to simulate comparatively novel ideas that evolves around DMM, such as control/data-plane split and functional distribution. This is because generic simulators do not facilitate such concepts. They are evolved around the general IP networks where control/data-planes are co-located at each component, which does not provide additional support required to implement those planes separately.

Another challenging task in the mobility protocol analysis is to confirm the performance in the real world scenarios. The performance of an MM scheme may highly depend on the topology underneath. For example, providing MM in a smaller network might be having least overhead and least complications. Thus, test results might not reflect the expected behavior of the scheme when employed in a real network. The real scenario might include a network which is extremely large and complex. This has led to a predicament in the DMM research community, where it remains extremely difficult to confirm performance of draft protocols. As a consequence, in spite of having a considerable amount of DMM ideas, standardization seems to lag behind. These factors urge the requirement for sustainable evaluation tools and a comprehensive study on already existing proposals to understand their applicability.

On the other hand, looking at the theoretical models of the proposed ideas, it is noticeable that they leave some unexplored areas where further enhancements can be made. For example, some proposals suggest fully-distributed DMM approach to provide node mobility, where both the data and control-planes are distributed. They require complicated consistency handling mechanism. Thus, it may result in a heavy controlling overhead. The proposals which suggest partial distribution where only the data-plane is distributed, do not address a single point of failure that may occur in the control-plane and the scalability limitations. None of these ideas have provoked optimality in both planes at the same time. Thus, it leaves a design space, where much prospective notions may lay concealed. Considering that the DMM is comparatively young, digging into such unplumbed areas may provide greater contribution to the DMM community.

1.4 Contributions

One of the main contributions of this thesis is the evaluation of existing proposals. It contributes the research community with a comprehensive study of DMM efforts found in the literature. Then, by identifying advantages and disadvantages of the already proposed ideas, the best approach to attain node mobility with DMM is expected to be identified.

Motivated by the non-existence of a proper comparison study on DMM and least favorable simulator support, a network layer (L3) simulator called SimNetDMM is designed and constructed. SimNetDMM is capable of exclusively analyzing DMM schemes. This is the initiative simulator seen in the literature which allows the evaluation of non-standard DMM schemes. Specially, it provides immense support to analyze DMM schemes which are rooted in novel concepts like control/data-plane split and fully-distributed control-plane. Thus, it is expected to benefit the DMM research community to a greater extend by contributing a specialized DMM simulator to analyze their future proposals.

Using SimNetDMM, draft DMM schemes found in the literature are evaluated in control and data-planes. This work is novel in the scope of DMM. There is no previous work focusing on performance evaluation and comparison of DMM schemes. Based on the results obtained, best approaches for DMM is recognized. Problems appear in the current proposals extending the best approaches are also recognized. Optimization of both control and data-planes is not confront by already proposed schemes. Thus, the vital necessity for a better solution is identified.

The next major contribution of this thesis is the introduction of the novel DMM scheme, DMMSDN. The results of the evaluation carried out emphasizes the capabilities of control-data plane split and fully distribution in the control-plane to achieve DMM. Based on the gaps that are detected in the draft proposals, DMMSDN is introduced. It is a fully distributed network-based DMM scheme rooted in control/data-plane separation as opposed to the work seen in the literature. This scheme utilizes both control / data-plane split and the fully distribution at the same time to achieve better performance in both control and data-planes. As a candidate to achieve this, the Software Defined Networking (SDN) paradigm is chosen. Moving away from replicated distribution of MMEs, DMMSDN focuses on the distribution of MM functionalities in the control-plane. This idea is novel in the control / data-plane split scope. Optimally distributing the control-plane with sustainable functional distribution is expected to improve the control-plane performance. Thereby, consistency handling is expected to be simplified,whereas the communication overhead between distributed MM Entities (MMEs) and the MME load is also expected to reduce. In order to confirm the applicability of DMMSDN, the performance of DMMSDN is compared against different DMM schemes. According to the results, DMMSDN achieves better performance in both control and data-planes. Finally, optimal distribution for economical DMM with DMMSDN is identified for different types of ISP networks.

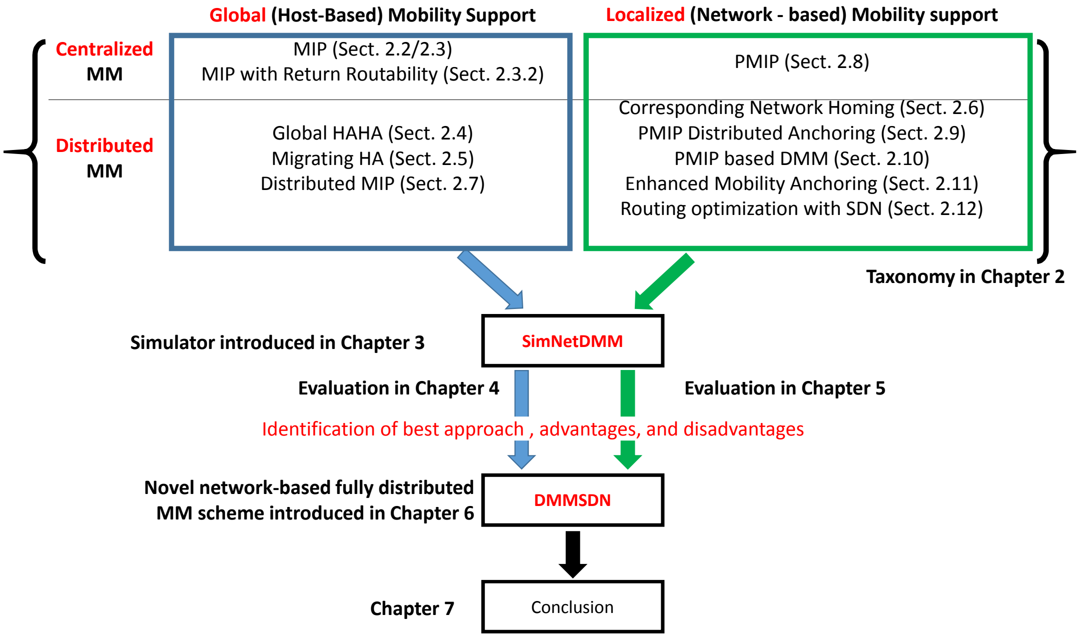

Global (Host-Based) Mobility Support
Localized (Network - based) Mobility support
Centralized MM
Distributed MM
MIP (Sect. 2.2/2.3)
MIP with Return Routability (Sect. 2.3.2)
Global HAHA (Sect. 2.4)
Migrating HA (Sect. 2.5)
Distributed MIP (Sect. 2.7)
PMIP (Sect. 2.8)
Corresponding Network Homing (Sect. 2.6)
PMIP Distributed Anchoring (Sect. 2.9)
PMIP based DMM (Sect. 2.10)
Enhanced Mobility Anchoring (Sect. 2.11)
Routing optimization with SDN (Sect. 2.12)
Taxonomy in Chapter 2
Simulator introduced in Chapter 3
SimNetDMM
Evaluation in Chapter 4
Evaluation in Chapter 5
Identification of best approach , advantages, and disadvantages
Novel network-based fully distributed MM scheme introduced in Chapter 6
DMMSDN
Chapter 7
Conclusion

Figure 1.11: Structure of the Thesis.

1.5 Structure of the Thesis

The composition of the rest of this thesis is given in Fig. 1.11.

Chapter 2 gives the taxonomy of MM schemes. First it introduces the main classifications that can be performed on MM attempts found in the literature. Then, it gives an account of significant MM attempts. Chapter 3 introduces SimNetDMM, which is constructed envisioning to evaluate DMM schemes. Design and the implementation of SimNetDMM is given in detail. Then, Chapt. 4 details the evaluation performed for global DMM schemes. This leads to the identification of best secondary HA placement in the MIPv6 environment for global DMM schemes. The conclusions are drawn from the perspective of the ISP. Then, the best approach to achieve global MM is identified. Chapter 5 introduces the evaluation that is performed over localized DMM schemes. It leads to identify the best approach to achieve the localized MM. Ultimately, it is recognized that each proposed scheme incorporates overheads either in controlling or data packet delivery. Therefore, the optimization of both control and data-planes is identified as candidate research area. A novel DMM scheme, known as DMMSDN is introduced in Chapt. 6 considering the above factors. Then, the performance of DMMSDN is confirmed by conducting two evaluations. One of the evaluations compares the performance of DMMSDN with several schemes found in the literature. The next evaluation examines the optimal SDN-controller distribution to attain optimal DMMSDN performance. Finally, Chapt. 7 concludes the thesis.

1.6 Summary

In the network layer, identity and the location of a certain node are distinguished using the IPv4 / IPv6 address of the node. This has led the nodes to loose the active sessions and reachability when it changes its location of attachment to the Internet. That is because it cannot continue using the previous IPv4 / IPv6 address at the new location. However, as the popularity for portable devices increases, users are trend towards mobility. Further, the mobile carrier networks have evolved towards packet switching based on IP. Thus, it is vital to provide session continuity and reachability for mobile nodes in the Internet. This is addressed by Mobility Management (MM).

Standard MM schemes employ a MM Entity (MME) in the home network of the MN, to which it keeps connected for a prolong time. This MME keeps the binding of the home address and the care-of address, which the MN obtains in the home and visited networks. When the MN is away from the home network, the MME forwards the traffic bound to the home address to the location specified by the care-of address. Thus, the incoming traffic is anchored at the MME.

However, these standard schemes suffer well-known problems. A single point of failure and attack, scalability limitations, non-optimal routing, and unnecessary traffic are a few of them. Thus, Distributed MM (DMM) was introduced. The idea of DMM is to distribute the MME within the network. Single point of failure and attack will be avoided due to the distribution. Moreover, by letting the MN to anchor at the closest MME, it is expected reduce non-optimal routing.

DMM is a comparatively young idea. No standardization has been made yet. Most of the DMM schemes are still in the proposal phase, which lacks prototyping, testing, and implementation. Thus, a requirement of a comprehensive study of DMM is identified. Simulation is identified to be the best approach to test such DMM schemes considering the real world scenarios like employment in the Internet or ISP networks. Further, due to lack of simulator support, it is recognized that a sustainable simulator is required to test non-standard protocols and new DMM concepts. Therefore, a suitable simulator called SimNetDMM is constructed and an extensive evaluation of DMM schemes found in the literature is conducted. Then, identifying advantages and disadvantages of the proposed ideas, it is determined that none of the existing proposals meet the optimal performance. Thus, a novel fully-distributed network-based DMM scheme utilizing control/data-plane separation, known as DMMSDN, is introduced. Finally, performance of DMMSDN is confirmed by conducting a set of evaluations.

Chapter 2

Taxonomy of Mobility Management Schemes

During the last couple of decades, there have been many attempts to address the MM in the Internet considering different aspects such as scope of mobility. This chapter provides a brief overview of the MM in the Internet. Then, it presents an account of a selected set of mobility management schemes found in the current MM context.

2.1 Network Layer Mobility Management

Mobility management attempts found in the literature are frequently categorized considering different aspects and properties they inherit [74]. Even though a well defined categorization does not exist, main characteristics that are conventionally used and referred in this thesis are listed below.

- **Scope of Mobility**, which takes into consideration the boundary defined for the mobility of the nodes. It can be boundless, allowing the node to roam within the Internet, or restricted to the local network.

- **Involvement of entities in MM** determines the modifications that are required in different entities to support MM. It may require modifications also to to be made to the host node or modifications can be limited to the network.

- **Placement of MMEs** conciliates sustainability and performance of the schemes. Different MME placement approaches may fall under *centralized*, *partially distributed*, or *fully distributed* categories.

- **Methodology to Provide IP Address Reachability** determines how the reachability of the IP address is facilitated. *ID / locator split* and *address delegation* serves as two main branches seen in IP address reachability provisioning. ID / locator split requires the management of ID to locator mapping, whereas address delegation suggests the delegation of IP address the newly attached network of the MN.

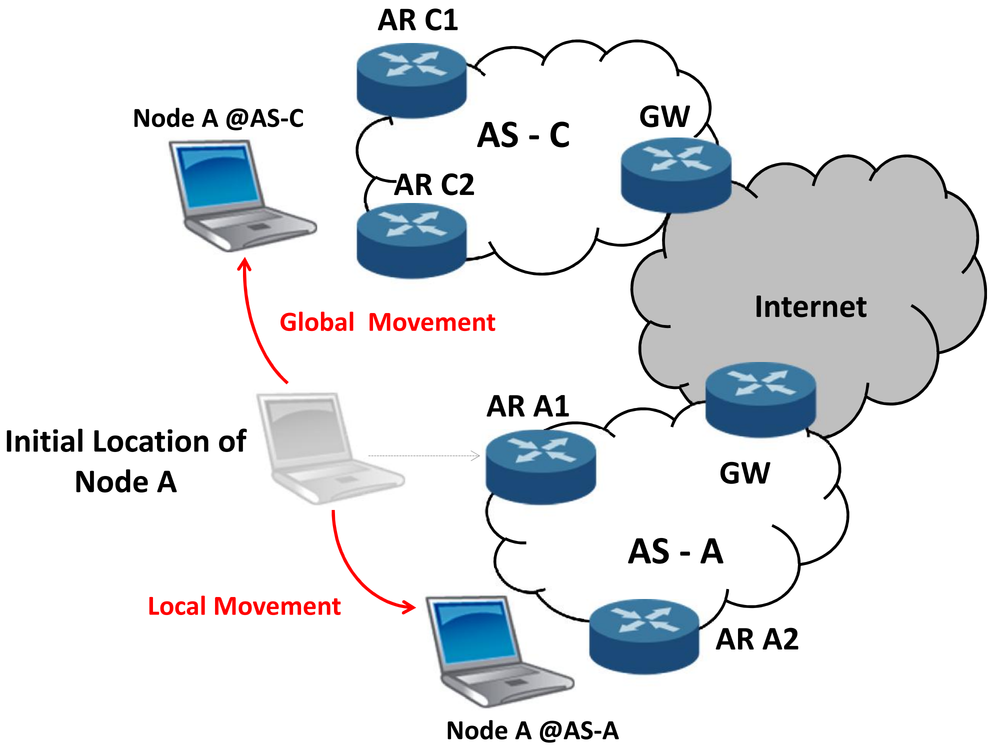


Figure 2.1: Scope of the Mobility.

- **Routing Method** includes *indirect routing* and *direct routing*, where it determines whether the CN is allowed to attain the MN's current location information.

Rest of this section describes those characteristic categories in detail, whereas Sect. 2.1.6 gives an insight to the selected MM schemes using above categorizations.

2.1.1 Scope of the Mobility

Scope of the mobility can be considered as a very important factor in the MM. It defines the extent to which the MN can move. *Local domain* is widely used to identify the ISP network to which the MN attaches to attain its IPV4/ IPv6 address. *Global mobility* allows the MN to roam within the Internet, without limiting to the local domain. On the other hand, *local mobility* is identified to be the scenario where the MN's movement is restricted to the local domain. In local mobility scenario, all the candidate ARs through which the MN might access the Internet are managed by the similar network administration. These two scenarios are depicted in Fig.2.1.

When *global mobility* is considered, it introduce a lot of complications. That is since the AR through which the MN reattaches to the Internet might not be a part of the local domain. This might lead to authentication and authorization complexities. Further, the sessions of the MNs are vulnerable to Internet threats such as denial of service attack, replay attack, address impersonation, and session hijacking. Thus, security measurements should be tight in global MM. Most antecedent MM attempts [18] [32][37] [50] [51] [61] evolved around the concept of global mobility.

In contrast, in *local mobility*, the mobility of the node is restricted to the local domain [33] [34]. Thus, the authorized and the authentication can be implemented easily since the MN does not leave the local domain. This can be considered as the main advantage of restricting the mobility of the node to the local scope, as location privacy is a major concern of ISPs. However, the concept of localized mobility was familiarized later to the literature [33] [34]. Thus, least standardizations exist [25].

2.1.2 Host-Based vs Network-Based Mobility Management

The two scenarios that are considered are roughly illustrated in Figs. 2.2 and 2.3.

The initial attempt to handle the mobility of nodes was considering the participation of MN in mobility handling (Fig. 2.2). The MN is assumed to detect the change in its attachment location to the Internet. Therefore,modifications in the MN is required in order to obtain the MM facility. The MN detects its movement when the MN is assigned a different IPv4 / IPv6 address than the one it used to have at the location where it is connected for a prolong time, known as home network. Then, the MN informs its current location to the MM entity which resides in its home network. Using this approach lets the MN to roam in the Internet without any restriction.

Apart from security issues and required modifications on the MN, the host-based approach has several other disadvantages as well. Usually the MNs might be connected to the network via wireless medium. Since the MN needs to participate in the mobility handling procedure, this wireless link might create a bottleneck. Earliest MM attempts [18] [35] [37] [50] [51] [61] which support global mobility are mainly host-based MM schemes.

On the other hand, network-based MM introduces a different approach to handle mobility of MNs. The above mentioned problems in the host-based MM can be considered as the inspiration in introducing network-based MM. If the mobility of the MN is restricted to a single domain, the network itself can detect and handle the mobility on behalf of the MN. As a result, it will avert the requirement to pass controlling signals between the network and the MN. In the most common scenario, the MN is expected to connect to the network via wireless links. It can be considered as a huge bottleneck. It is due to the fact that wireless links have lower bandwidths with much congested conditions. As a consequence, if the number of signals passed between the MN and the network is significant, then the time it takes to perform the handover can dramatically increase. This might result in a considerable packet loss. Thus, when the mobility handling is pushed to the network, it improves handover delay. There are a handful number of attempts that address network-based MM in the literature [25] [32].

Most importantly, the network-based MM enables legacy MNs to benefit from MM, since it does not require any modification in the MN. Further, it assures location privacy of the MN. It is as a result of not assigning a new IPv4 / IPv6 address to the MN at the new location. Thus, the topology would be not exposed.

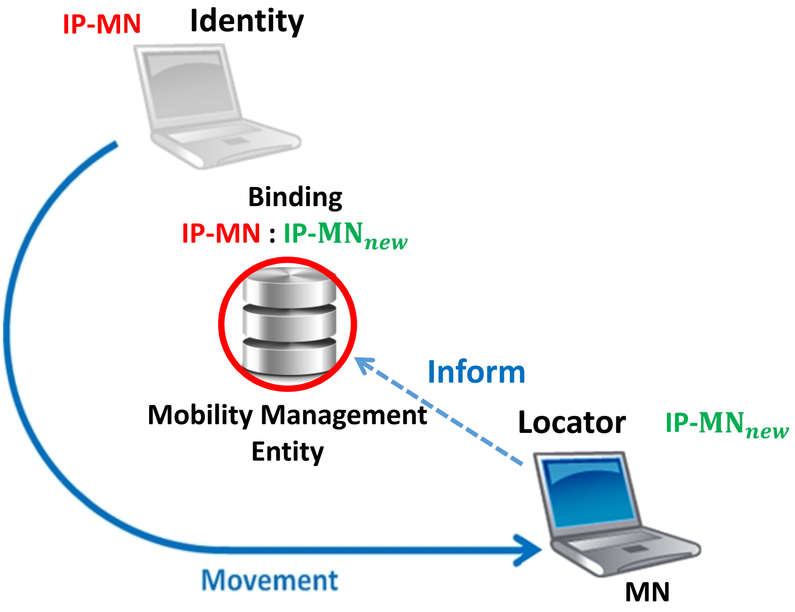


Figure 2.2: Host-based Mobility Management.

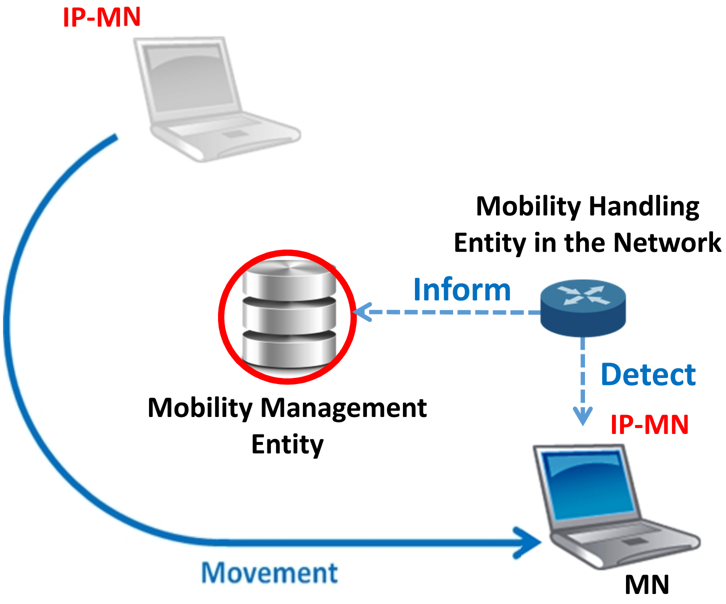

IP-MN
Mobility Handling Entity in the Network
Inform
Detect
IP-MN
Mobility Management Entity
Movement
MN

Figure 2.3: Network-based Mobility Management.

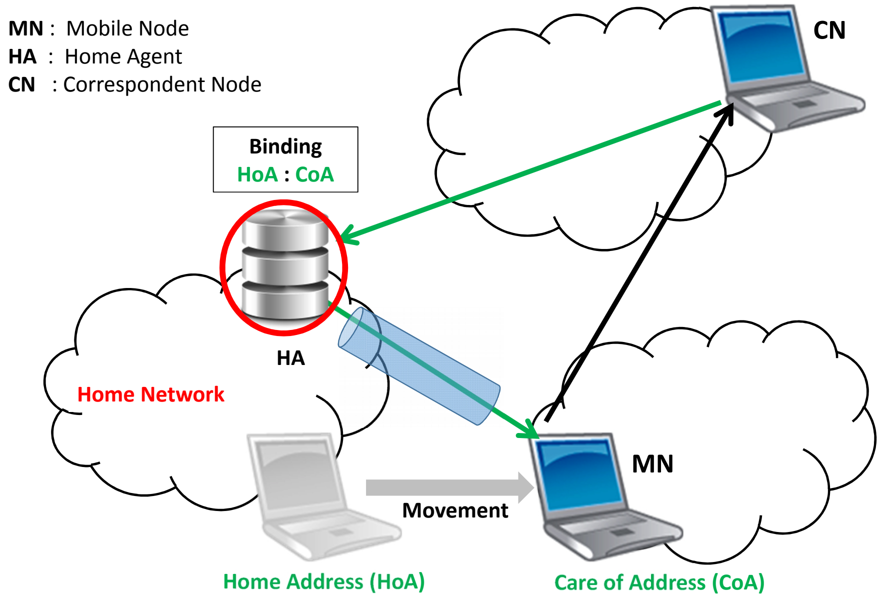


Figure 2.4: Centralized Control-plane.

This can also be considered as an advantage of providing network-based MM. However, as disadvantages, the mobility of the MN will be restricted to the local domain of the network and the network should be modified considerably to support mobility.

2.1.3 Centralized vs Distributed Mobility Management

Centralized MM (CMM) was the very first MM approach that was adopted. It introduces a single entity to handle mobility on behalf of the MNs. Figure 2.4 shows an instance of centralized MM. It assures that the mobility information is consistent as the MN moves. As advantages of this approach, the autonomy can be considered. It does not have any communication or dependent MM decision making. Thus, least control overhead is expected. Further, anchoring location is static, thus, the implementation is simple. Literature contains plenty of standard centralized MM schemes [18] [37] [50] [51] [61].

However, as already discussed in Sect. 1.2.2, centralized MM suffers several detriments. A single point of failure and attack, scalability issues, and non-optimal routing can be considered as a few of the well know problems.

Distribution of MM rather than relying on a single centralized MME is the essence of DMM. MM has become an essential functionality in the Internet recently, due to the enormous use of portable devices to access the Internet. This demands scalability and better network performance with MM. DMM allows to meet this demand by suggesting a much sustainable approach. When the scope of distribution is considered, it can be of two types.

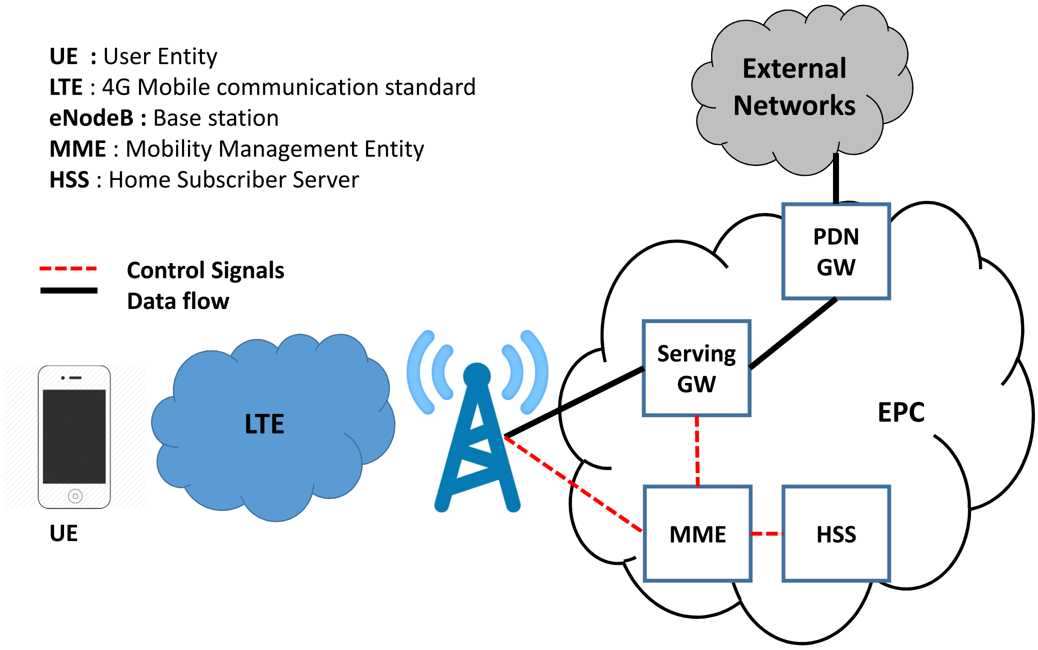


Figure 2.5: Architecture of Control/Data-Plane Split Driven EPC.

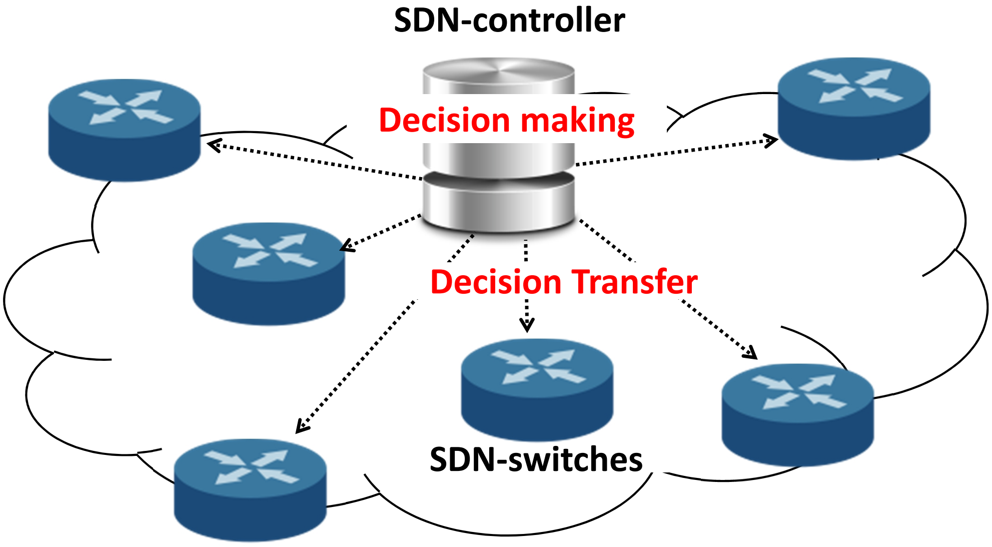


Figure 2.6: Control/Data-Plane Separation with SDN.

- Partially distributed
- Fully distributed

The idea of partially / fully distributing the MM is rooted in the idea of control/data-plane separation. In network layer MM protocols, it is identified that there are two types of packets that are exchanged. One is the exact data packets that should be exchanged between two nodes. The other is the packets that are exchanged for the controlling purposes. For instance, setting up the MM and information about location of the nodes can be considered as two of the many such controlling related packets. Thus, the control and data-planes are considered to be uniquely facilitating the control information delivery and data delivery purposes, respectively. However, the current network architecture of the Internet does not implement these two specific planes separately. Thus, they are established as a single unit. For example, routers in today's Internet perform both control and data-plane processing. Nonetheless, recent advancement in mobile carrier networks towards Evolved Packet Core (EPC) [2] has successfully employed the control/data-plane separation. Thus, it limits the data-plane to the functionality of forwarding data only, whereas the control-plane involves only in controlling procedure. This idea is depicted in Fig. 2.5.

Further, the Software Defined Networking (SDN) [1] paradigm proposes the control and data-plane separation where the control-plane is centralized in the SDN-controller. An overview of the SDN paradigm is given in Fig. 2.6. Data-plane is implemented in SDN-switches, where they simply perform the data packet forwarding as determined by the centralized SDN-controller. Thus, the control-plane draws mobility related decisions and data-plane only forwards the data traffic accordingly.

2.1.3.1 Partially Distributed Mobility Management

In partially distributed MM, it only distributes the data-plane. Partially distributed MM keeps the control-plane centralized, thus it assures the autonomy of MM. Then, by distributing the data-plane, it tries to reduce non-optimal routing. That is by bringing the data anchoring entity closer to the MN. Figure 2.7 shows an example of a scheme that depends on this approach. This scheme utilizes the SDN paradigm to achieve control-plane centralization and data-plane distribution.

2.1.3.2 Fully Distributed Mobility Management

Fully distributed MM extends the above described partially distributed MM by distributing the control-plane as well. By doing so, it tries to avoid the single point of failure that can occur in the control-plane. Further, the distribution of decision making entity will reduce the load handled by each MME. Figure 2.8 illustrates an example scenario of fully distributed MM [67] [68]. The HA is distributed in the Internet, where any HA is capable of receiving / updating bindings and maintain anchoring. However, fully distributed MM may require a distinctive effort in keeping the consistency and achieving autonomy.

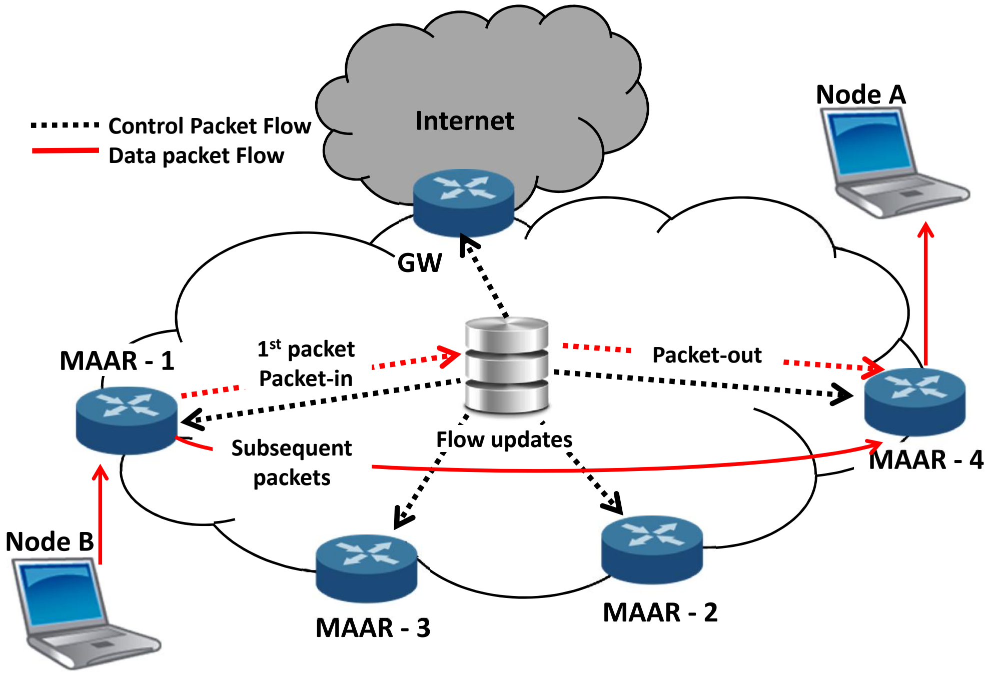


Figure 2.7: Partially Distributed Mobility Management.

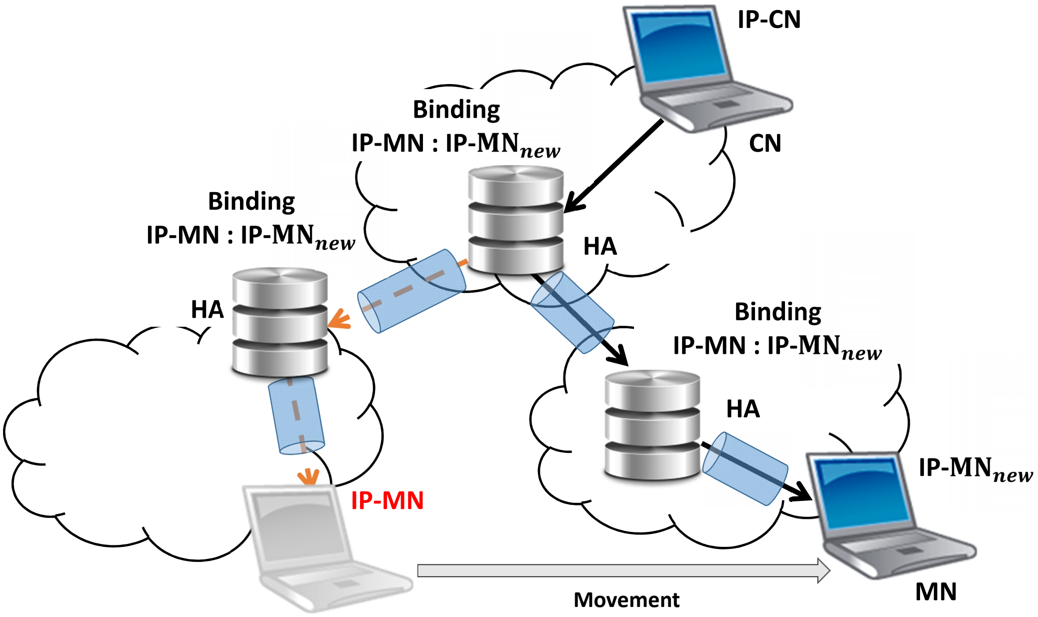


Figure 2.8: Fully Distributed Mobility Management.

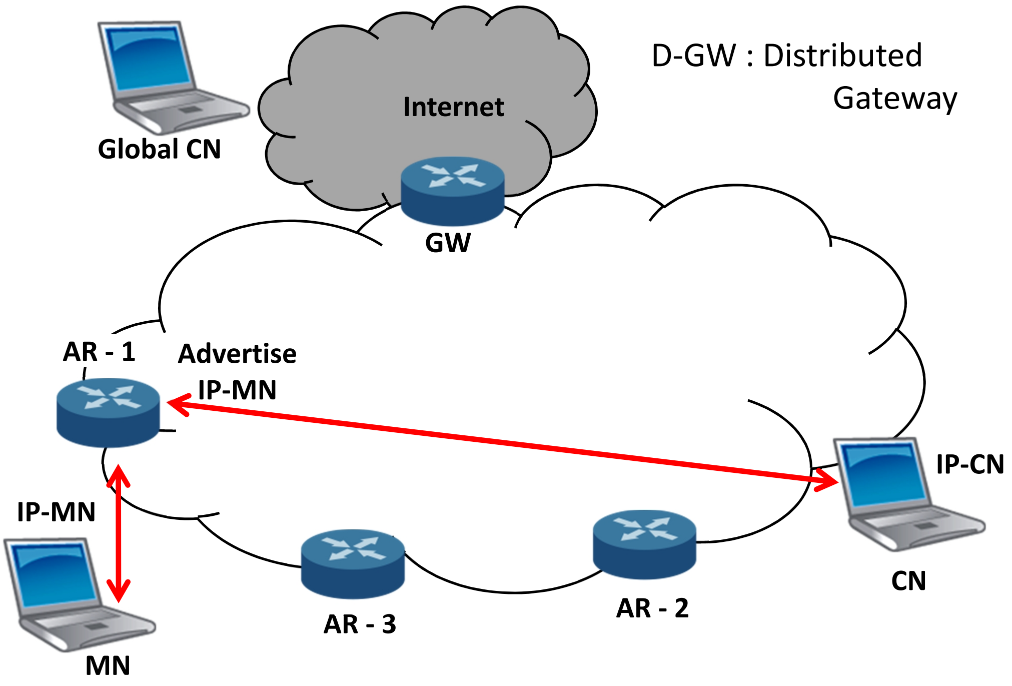


Figure 2.9: Functionality of Address Delegation - Intialization.

2.1.4 Methodology to Provide IP Address Reachability

When DMM efforts in the literature are investigated, two branches of address allocation and reachability provision methodologies can be identified. One can be considered as allow the MN to have different IPv4 / IPv6 addresses to determine its *locations* and an *identity* which is location independent. Then, the correspondence between those two, known as *binding* is maintained. It allows the sessions that MN started using the *Identity* to be alive without disruption, regardless of the IPv4 / IPv6 address changes. This does not introduce any overhead on other network layer protocols or other layers. However, achieving routing optimization with this methodology might be extremely costly. That is due to the fact that the correspondence between the *identity* and the *locator* should be resolved closer to the network edge in order to yield optimal data-plane performance.

The next methodology which requires IPv4 / IPv6 address delegation requires special modifications in routing tables if it is applied in the current network architecture. The basic functionality is illustrated in Figs.2.9 and 2.10. This methodology assures the optimal routing naturally. However, it might require the whole network to converge in order to attain consistency. Thus, based on the size of the network and the adopted routing table update methodology, performance of this approach may vary. Further, if the convergence time is too long, it might result in packet loss.

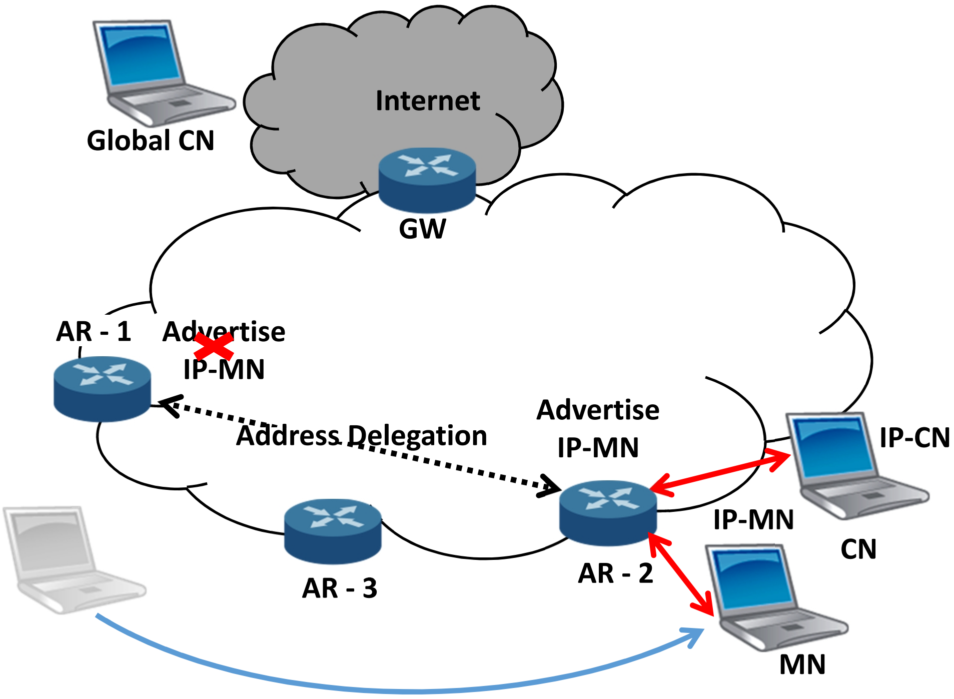


Figure 2.10: Functionality of Address Delegation - Handover.

2.1.5 Routing Mode

In general, MM schemes adopt two routing modes in routing packets to the MN. The first one is known as *indirect routing*, which is encountered more frequently [50] [51] [25]. As the name implies, the packets are routed to the MN following an indirect approach. This is shown in Fig. 2.11. In this approach, the packets destined to the MN is captured by some entity and the entity encapsulates the packets and sends it to the new location. This is known as tunneling. Thus, it requires a special mechanism to decapsulate these data packets. It can be taken place at the MN or at a network entity which handles mobility on behalf of the MN.

In the *direct routing* mode (Fig. 2.12), the CN and the MN are allowed to communicate directly, even after the relocation of the MN ([50]- Return Routability). This does not require an interference of an MME when delivering data packets. This mode can be achieved using the previously discussed address delegations method. Otherwise, in order to achieve this mode, not only the MN, but also the CN should have special features. Those features should allow the MN to notify all the CN with which the MN is willing to continue sessions of its new location. On the other hand, the CN should be able to respond to the notification by altering the way the CNs sessions address the MNs. This introduce a lot of security risks as well as high signaling complexity.

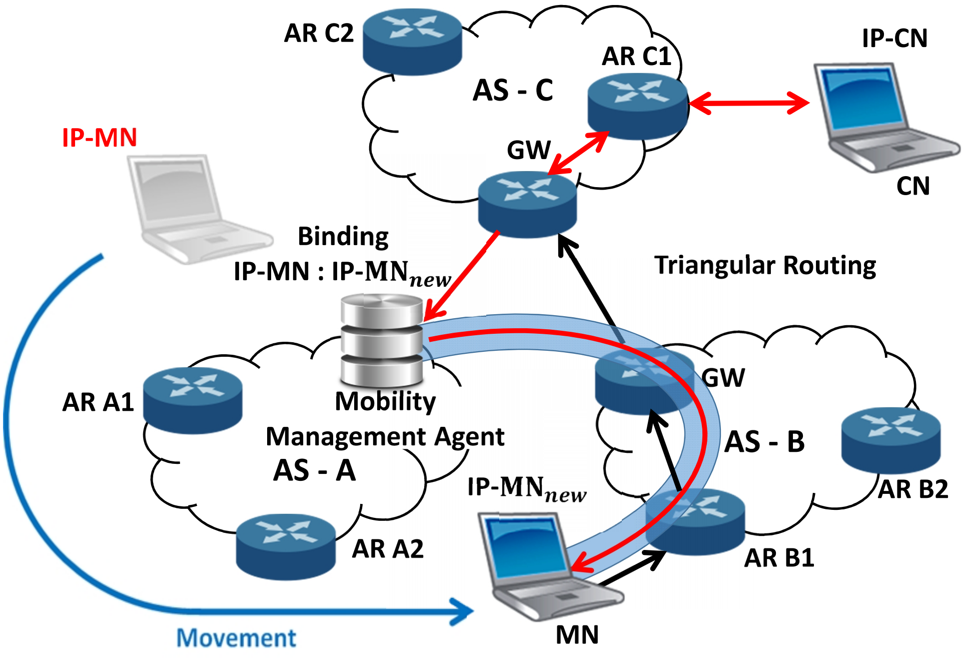


Figure 2.11: Indirect Routing.

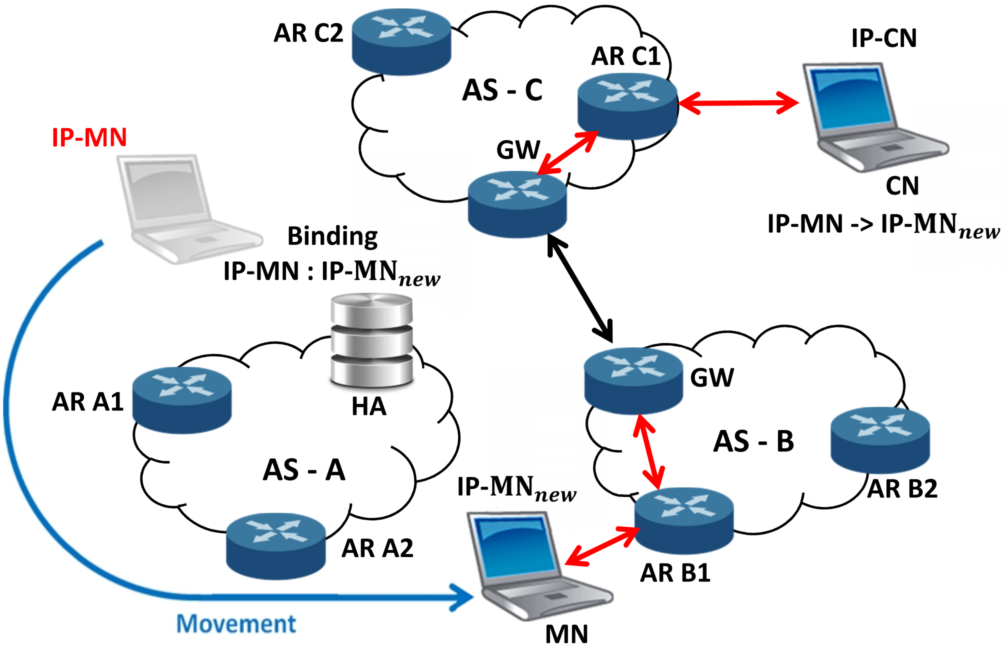


Figure 2.12: Direct Routing.

2.1.6 Mobility Management Scheme Categorization

Table 2.1: Properties of Mobility Management Schemes - I (Global).

Property		MIPv4 (Sect.2.2) [50]	MIPv6 (Sect.2.3) [51]	GHAHA (Sect.2.4) [68]	MHA (Sect.2.5) [67]	DMIP (Sect.2.7) [15]
Scope	Global	●	●	●	●	●
	Local					
MM Trigger	Host	●	●	●	●	●
	Network					
Distribution	Central	●	●			
	Partial					
	Full			●	●	●
IP Addr. Alloc.	ID/ Loc.					
	Addr. Dele.					
Routing Mode	Direct		RR			
	Indirect	●	●	●	●	●

Table 2.2: Properties of Mobility Management Schemes - II (Localized).

Property		PMIPv6 (Sect.2.8) [25]	Extended PMIPv6 (Sect.2.9) [7]	DPMIPv6 (Sec.2.10) [36]	Address Delegation (Sect.2.11) [16]	RO-SDN (Sect.2.12) [72]	CNH (Sect.2.6) [73]
Scope	Global	●	●	●	●	●	●
	Local	●	●	●	●	●	●
MM Trigger	Host						●
	Network	●	●	●	●	●	
Distribution	Central	●	●				●
	Partial				●	●	
	Full			●			
IP Addr. Alloc.	ID/ Loc.						
	Addr. Dele.				●	●	
Routing Mode	Direct				●	●	
	Indirect	●	●	●			●

MM schemes in the literature are tend to inherit the about mentioned properties in different degrees. Tables 2.1 and 2.2 give an extensive overview of the MM schemes introduced in this chapter and the properties they inherit. This can be used as a guildline to understand the performance they are expected to yeild as well.

The rest of this thesis forcus on the global and local mobility. Thus, the main categorization that is considered is the global and local schemes. Table 2.1 lists the global MM schemes whereas Table 2.2 lists the localized MM schemes.

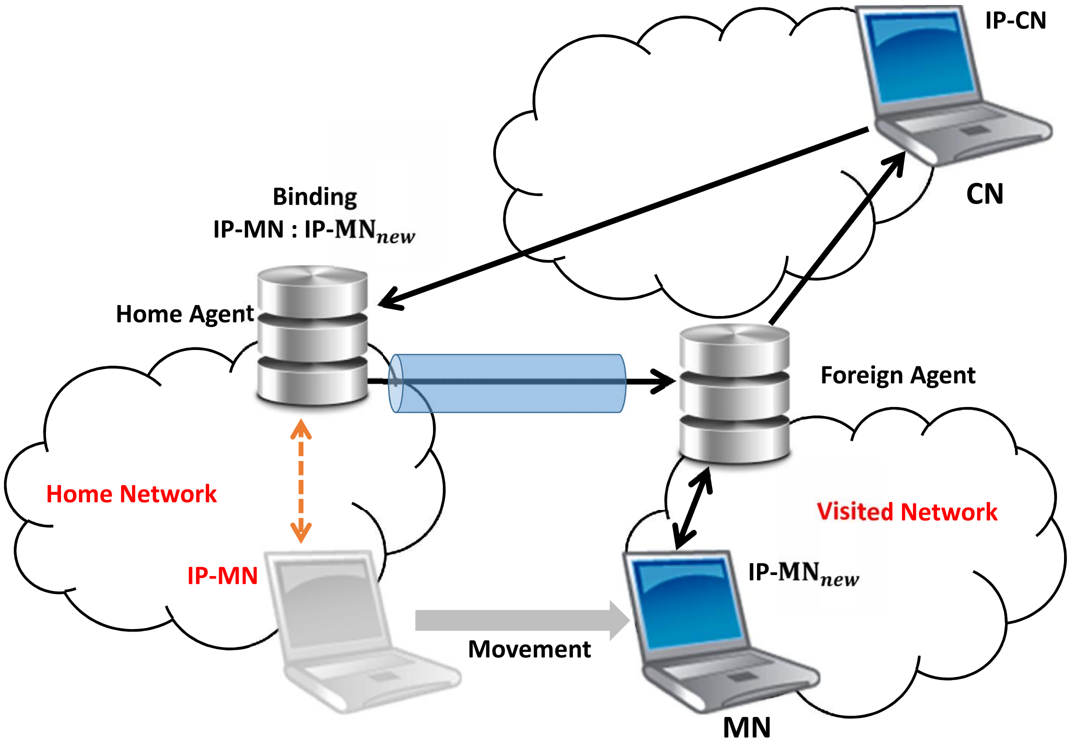


Figure 2.13: Overview of MIPv4.

2.2 Mobile IPv4

Assume that the MN attaches to the Internet initially through its default network, which is identified as the *home network* in the MM scope. Then, the MN will be assigned an IPv4 address, known as the *Home Address* (HoA). In general, as the packet routing mechanism in the Internet uses the destination IPv4 address to find the location to deliver the packet, those packets which are destined to the HoA will always end up at the home network. The initial idea to address the IPv4 mobility was born around this concept. It is widely known as Mobile IPv4 [50]. An agent is introduced in the home network, which is identified as the *Home Agent* (HA). The HA is capable of intercepting the packets arriving at the home network that are destined to the HoA.

MIPv4 introduces another agent called the *Foreign Agent* (FA), which resides at the network which is visited by the MN. This network is known as the *foreign network*. MN obtains the IPv4 address called *Care-of Address* (CoA) from the FA, when it attaches to the foreign network. Then, MN should inform this CoA to the HA to enable mobility. This CoA is usually the IPv4 address of the FA.

Assume that the Correspondent Node (CN) tries to send packets addressing MN using the HoA. Then, when the packets destined to the MN arrives at the home network, they will be intercepted by the HA and will be tunneled to the FA using the CoA. Then, the FA decapsulates the packets and forwards them to the MN.

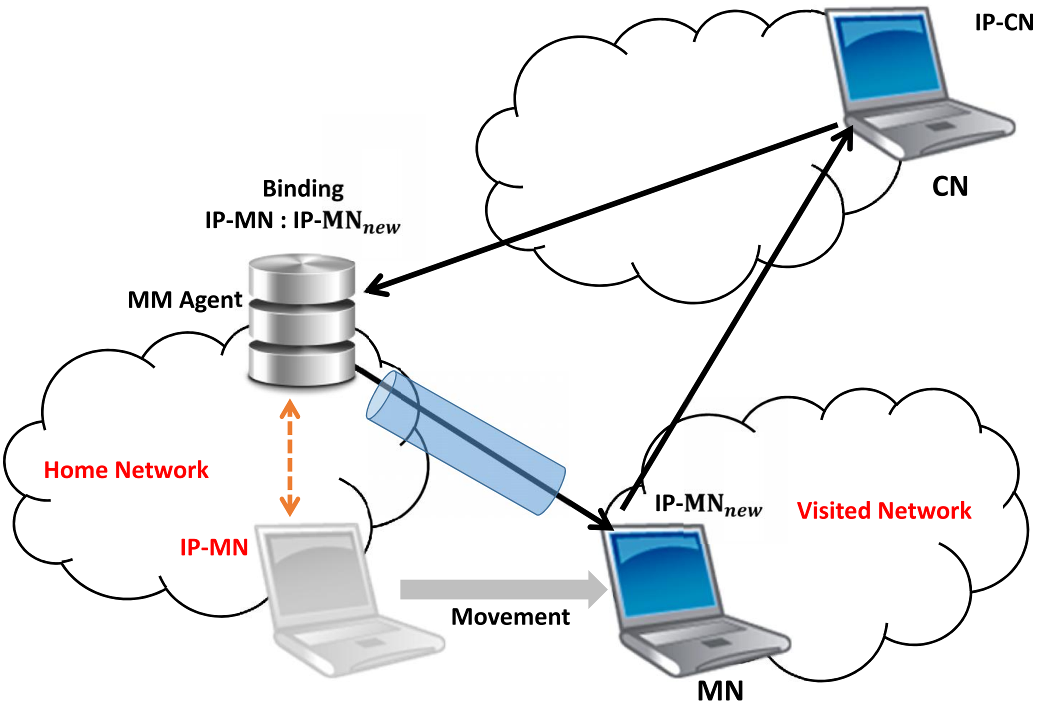


Figure 2.14: Overview of MIPv6.

The brief overview of MIPv4 is given in Fig. 2.13. The study given in this thesis focuses on the IPv6 environment. Thus, MIPv4 is given only as a reference protocol.

2.3 Mobile IPv6

Mobile IPv6 (MIPv6) [51] [74] adopts Mobile IPv4 to the IPv6 environment with slight modifications. MIPv6 adopts the concept of HA, similar to MIPv4. It resides in the network where MN is identified to be connected for a prolonged time, known as *home network*. HoA is used to identify the MN regardless of its later attachment locations, which is exactly the same as MIPv4. However, it does not employ an FA in the visited network. When the MN attaches via a different location, the MN obtains its new IPv6 address at the visited network. Then, the *Binding Update* (BU) is sent to the HA by the MN. This BU includes MN's new IPv6 address, which serves to locate the MN at its new location. This new IPv6 address which is assigned to the MN at its visited network is referred to as CoA. The HA will accept the binding and reply the MN with the Binding Acknowledgement (BA). The HA holds responsibility of re-routing data packets and facilitating continuity for ongoing communications by maintaining the binding between the HoA and the CoA. Figure 2.14 shows the data flow between CN and MN at the HoA and CoA of MIPv6.

Based on the implementation, two different routing modes can be identified. One

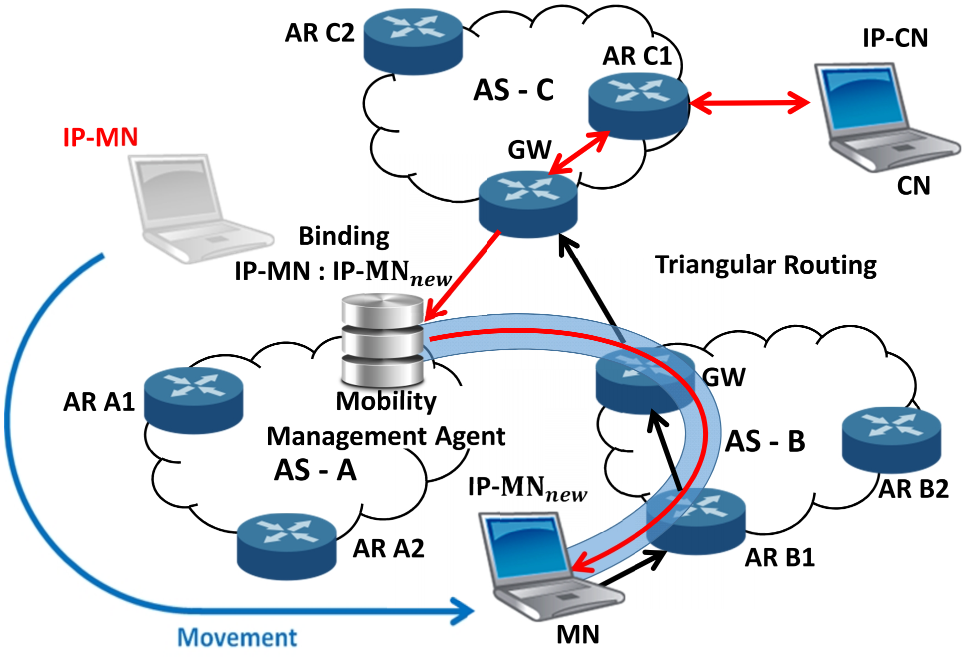


Figure 2.15: Indirect Routing.

is *indirect routing*, where the communication between the CN and the MN is mediated by the HA using a tunnel between the HA and the MN. Thus, all the packets will have to make a detour. If the CN does not support MIPv6, then this tunnel is a bidirectional tunnel. Otherwise, the tunneling is only for data packets en-route to MN. The next routing mode is known as *direct routing*. In this mode, interference of the HA is avoided and the general routing procedure is adopted. This lets the CN to use the CoA of the MN, rather than using the HoA. In order to allow the CN to obtain the CoA, this procedure introduces a complicated signalling procedure called return routability. As a result, the CN learns MN's CoA securely.

2.3.1 Indirect-Routing with MIPv6

Figure 2.15 specifies the data flow during *indirect-routing*. Assuming that the CN does understand MIPv6, the HA establishes a tunnel for data packets en-route to MN. Then, the packets destined to the MN is intercepted by the HA, when they arrive at the home network. Using the pre-established tunnel, then the data packets are forwarded to the MN. This results in non-optimal routing for the packets en-route to the MN. As a result, this architecture suffers triangular routing.

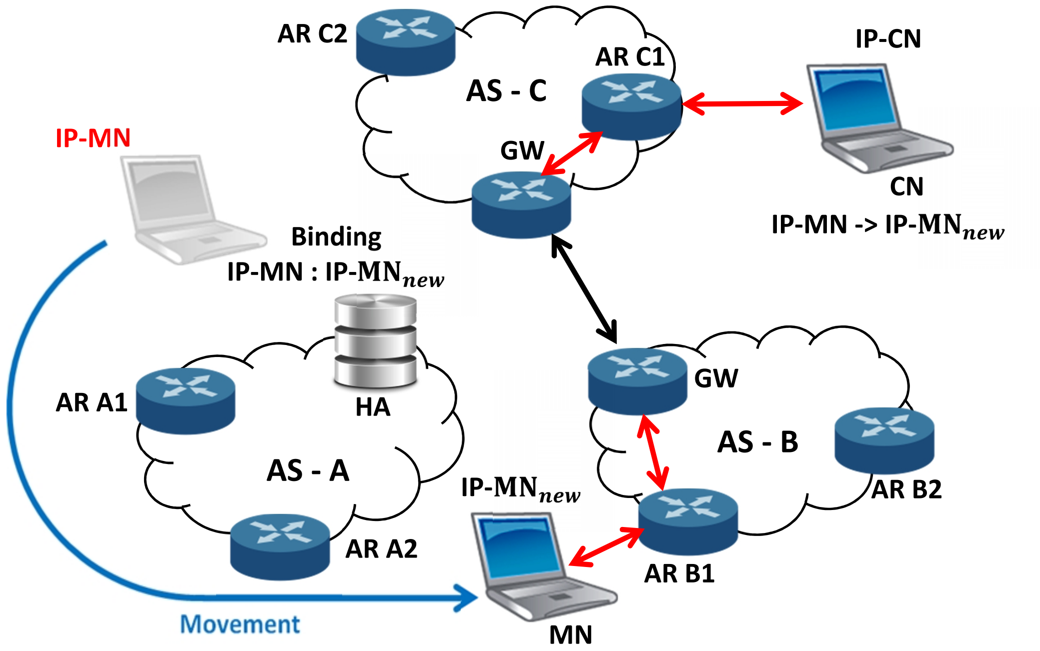


Figure 2.16: Direct Routing with Return Routability Procedure (MIPv6).

2.3.2 MIPv6 with Return Routability Procedure (MIPv6RR)

To overcome the burden of triangular routing resulted by indirect routing, the Return Routability (RR) procedure was introduced addressing security concerns with additional signaling to CNs that support this routing procedure. This scenario is shown in Fig. 2.16. Since it requires special support from CN, this mode is available only with non-legacy CNs which support this routing mode. Comparing it against general MIPv6, it requires additional signalling in order to allow the CN to obtain authentic CoA of the MN. Thus, following four messages are introduced in addition to the general MIPv6 controlling messages.

1. Home Test Init (HoTI)
2. Care-of Test Init (CoTI)
3. Home Test (HoT)
4. Care-of Test (CoT)

Control sequence of the MIPv6RR is given in Fig. 2.17. This procedure, assures secured route optimization by notifying the MN’s CoA to the CN. Following that, the CN can directly send the packet to the MN’s current location without requiring the HA to intercept and reroute the packets towards the MN as in general MIPv6. However, this requires MN to update not only the HA during the relocation, but also all the CNs with

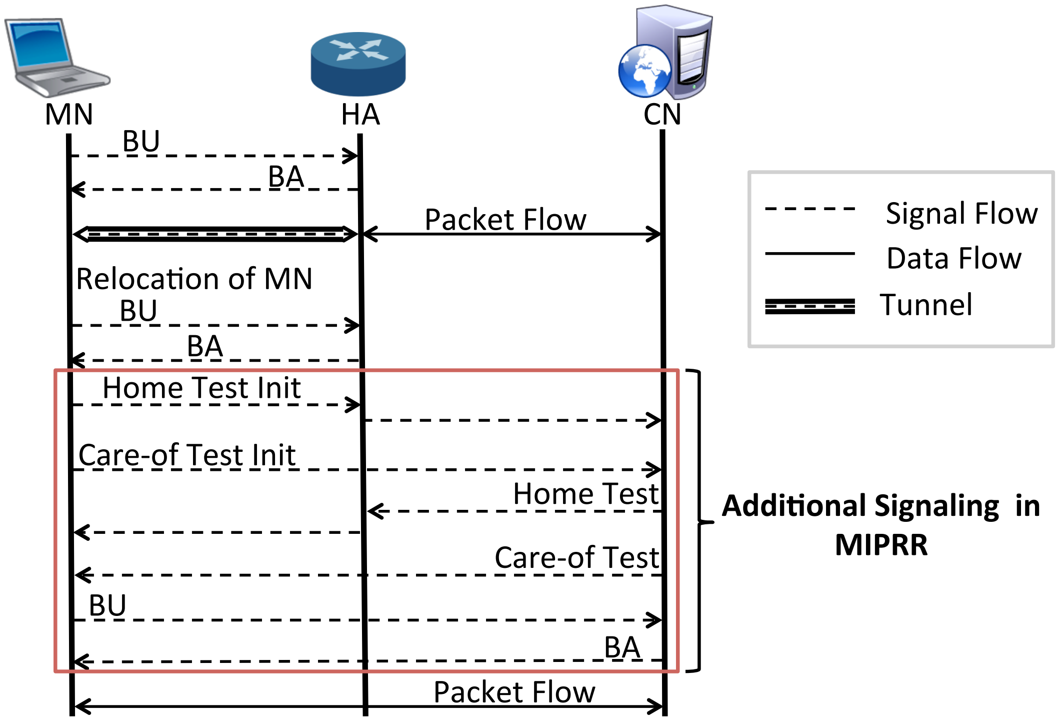


Figure 2.17: Control and Data Flow of MIPv6RR.

ongoing sessions, those support Return Routability procedure. Apart from that, it also introduces some security issues since the new location of the MN is being exposed to CNs. Traffic overhead also can be considered as an unfavorable factor of MIPv6RR.

2.4 Global Home Agent to Home Agent (Global HAHA)

MIPv6, which has a central MM entity, is inclined to suffer from a single point of failure and attack. More to the point, triangular routing and resistivity to scale-up are significant among major complications of MIPv6. As described in the previous subsection, although the routing optimization can be achieved by the Return Routability procedure, it has several complications. It is as a result of the presence of incompatible legacy devices and added signaling complexity. Global Home Agent to Home Agent (Global HAHA) [68] was introduced aiming to overcome these issues by adapting a DMM approach. Whereby, it reduces the risks bounded by single point of control.

Global HAHA proposes the employment of an overlay of HAs instead of a single HA. Thereby, it attempts to bring the HA closer to the MN. This is expected to reduce the Round Trip Time (RTT) between the HA and the MN. Then, it will lead to reduce the handover time of MNs residing farther from the HA. Each HA in the HA overlay is forced to advertise the same home prefix. Further, they are assigned two types of IPv6 addresses. An anycast IPv6 address is assigned to the HAs, which allows any HA to intercept any incoming packet addressing a prefix belonging to the common home prefix. Apart from that two unicast addresses, one assigned by the home prefix termed as *HA unicast address*, and the other assigned by the ISP which is the *HA locator address*, are associated with each HA. The *HA unicast* address is used by the MN when it wants

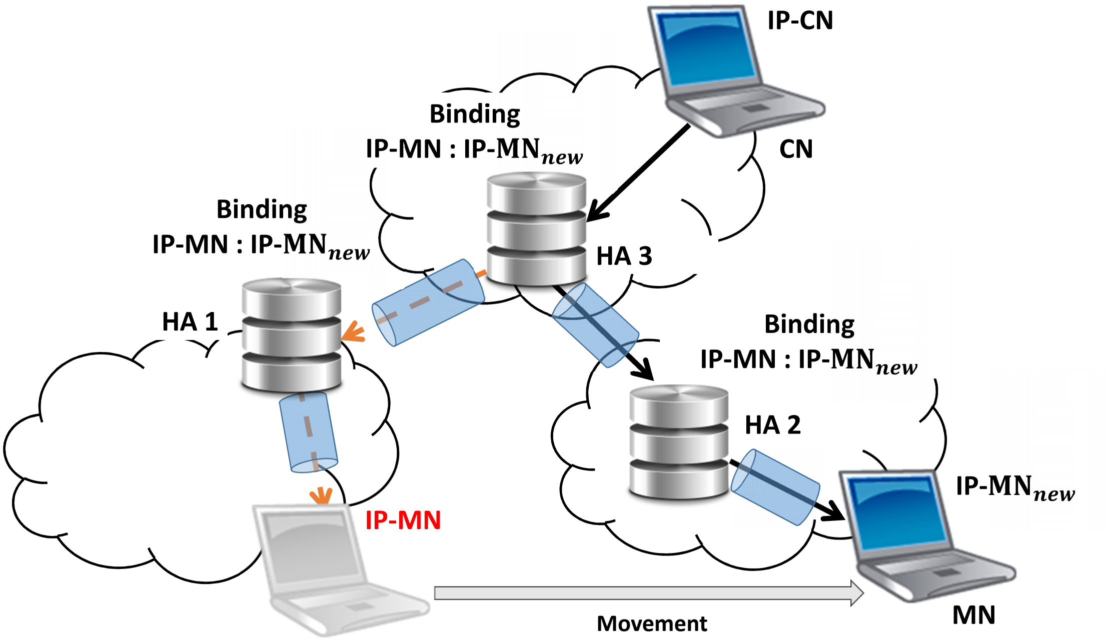


Figure 2.18: Overview of Global HAHA.

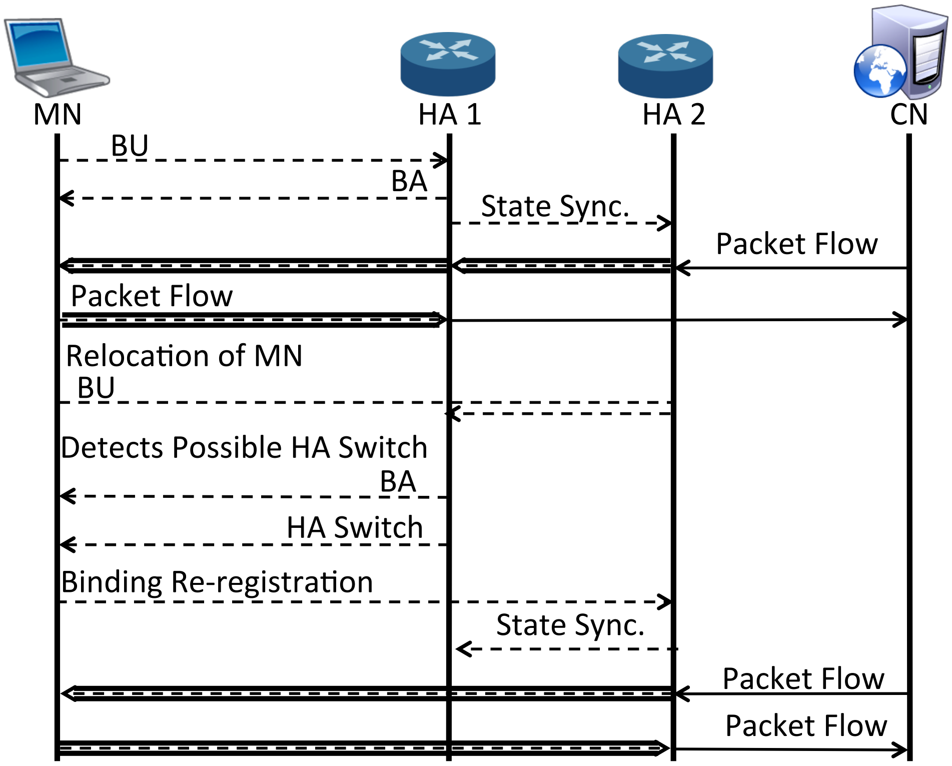


Figure 2.19: Control and Data Flow of Global HAHA.

to send BU to its HA. When the HAs exchange mobility signals, they use the HA locator address which specifies the actual location of the HA. The basic functionality of Global HAHA is detailed in Figs 2.18 and 2.19.

First the MN registers with an HA, by sending an anycast BU where the closest HA will receive it and respond. In order to maintain consistency, whenever any HA responds to a BU, it updates all the HAs on the new binding by using internal updates. This is due to the fact that all the HAs maintain binding details of all the MNs. When the MN changes its location, it sends an anycast BU towards the HA group as in MIPv6. This will be received by the closest HA, as all the HAs advertise the same home prefix. At that point, if the BU is forwarded to the current HA of the MN via another HA, a possible HA switch is notified back to the MN. That is because the interception of the second HA means that the second HA is more closer to the MN than the previous one. The previous HA notifies the MN about the closest HA by sending an *HA switch* message embedding the unicast address of the second HA along with the BA. Triggered by this, the MN will resend the *binding re-registration* message request to the closest HA. Whichever the HA that maintains the current binding with the MN, becomes the *primary HA*. Thus, the primary HA selection is dependent on each attachment location of the MN. However, they all advertise the HoA on behalf of the MN.

Similarly, the inward traffic bounded to MN, which is initiated by CN, will also be intercepted by the HA closest to CN. Then, since that HA knows which of the HAs among the group possesses the binding (primary HA), it will redirect the packet flow towards the primary HA of the MN which will then send the packets to the MN. As opposed to MIPv6, Global HAHA makes sure that the HAs involved in packet forwarding are the closest to the MN and the CN. Thus, non-optimal routing is expected to reduce up to some extent. One of the challenges of this solution is maintaining consistency of replicas of MM information, which is costly in terms of controlling.

2.5 Migrating Home Agent

Migrating Home Agent (MHA) [67] is an enhancement suggested to the Global HAHA scheme (Fig. 2.20). It suggests the replication of HA in an overlay exactly in the same way as Global HAHA by employing an overlay of HAs. In Global HAHA, the MN is required to de-register itself from the HA it was registered with, before registering with a different HA, which is closer to the MN than the previous one. Thus, it has introduced some controlling overhead between MN and the previous HA. Theoretically, if a de-registration takes place, the involved MN and HA are lying farther from each other. That it the reason why a HA switch takes place. Thus, aiming to reduce the overhead occurred during the HA switch procedure introduced in the Global HAHA, MHA suggests the handling of HA switch at the network end. This is intend to reduce the control over head of HA switch in Global HAHA.

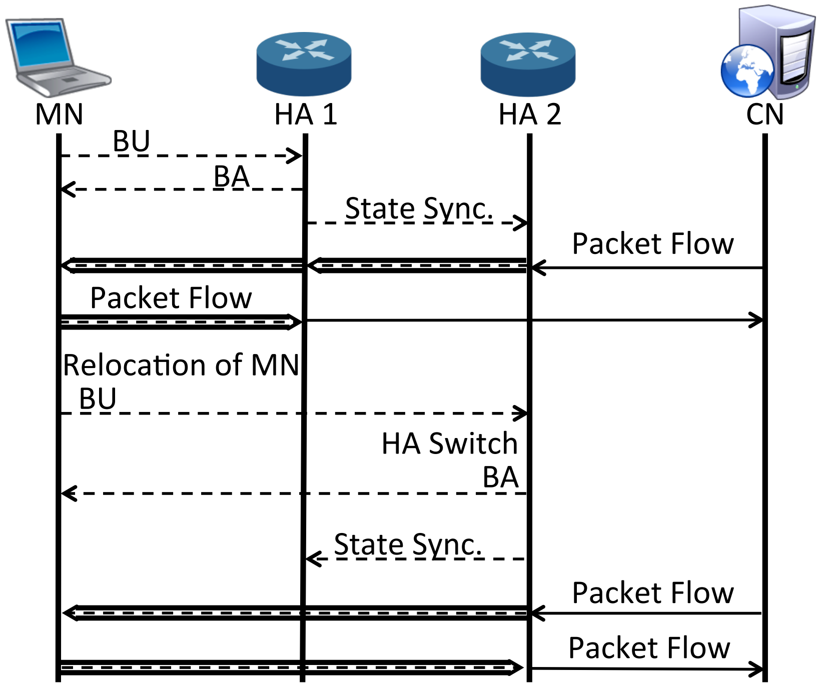


Figure 2.20: Control and Data Flow of Migrating HA.

2.6 Corresponding Network Homing

The major drawback seen in the MIPv6 can be identified as the placement of the HA in the local network of the MN. It might cause a severe non-optimal routing scenario. Thus, Corresponding Network Homing (CNH) [73] suggests allowing the MN to anchor closer to the CN. This scheme suggests that most of the time, MNs tend to create sessions with stationary nodes. So, as opposed to placing the HA in MN's *local network*, the CNH scheme suggests that having the HA near the CN will reduce the non-optimal routing.

However, when the CN is not a part of the MN's *local network*, CNH scheme requires the network operators who facilitate mobility to come to a negotiation with respective CN networks. Thus, CNH can be seen as a rather application oriented solution, where a business agreement with the CN networks may allow the MN to maintain session continuity. Figures 2.21 and 2.22 illustrates the general functionality of the CNH scheme during initialization and handover. Further, the control and data flows are shown in Fig. 2.23.

The HA installed at the CN's end is known as the Corresponding Home Agent (CHA). Some cases, the CHA can be co-located with the CN itself. However, it is not essential for the CN's network to host the CHA. In contrast, a third party can host the CHA. Thus, the only requirement for the CHA is to be closest to the CN.

When an application residing in the MN tries to initiate communication with a specific CN, the MN resolves the address of the CN. Additionally, it tries to discover the CHA that serves the CN. First, the MN sends the BU message to the CHA requesting a dynamically-allocated Corresponding Home Address (CHoA). Once when the CHA receives the request, it assigns a CHoA and establishes a tunnel between the MN's current location and the CHA. The data flow will always be intercepted by the CHA.

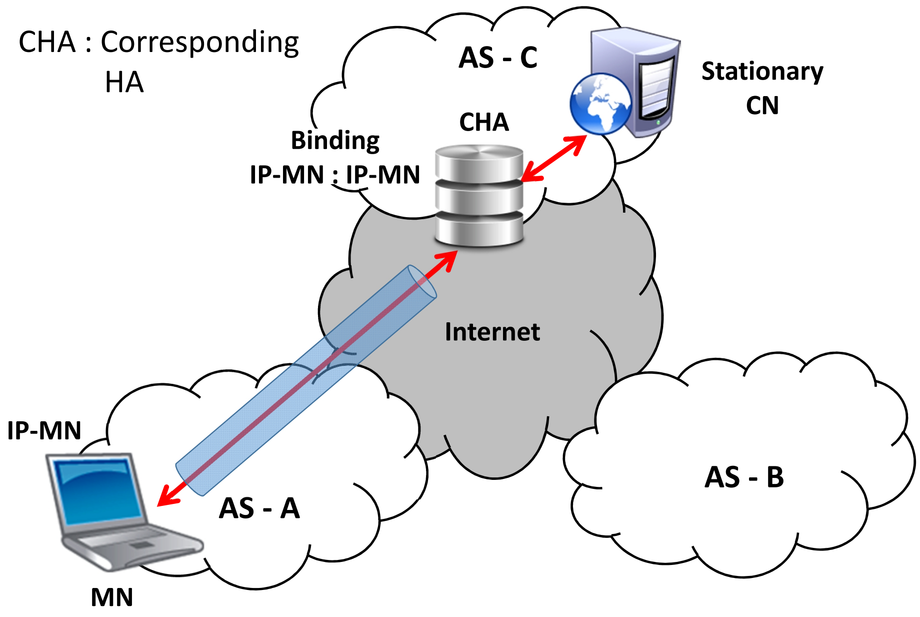


Figure 2.21: Corresponding Network Homing Scheme - Initial Location.

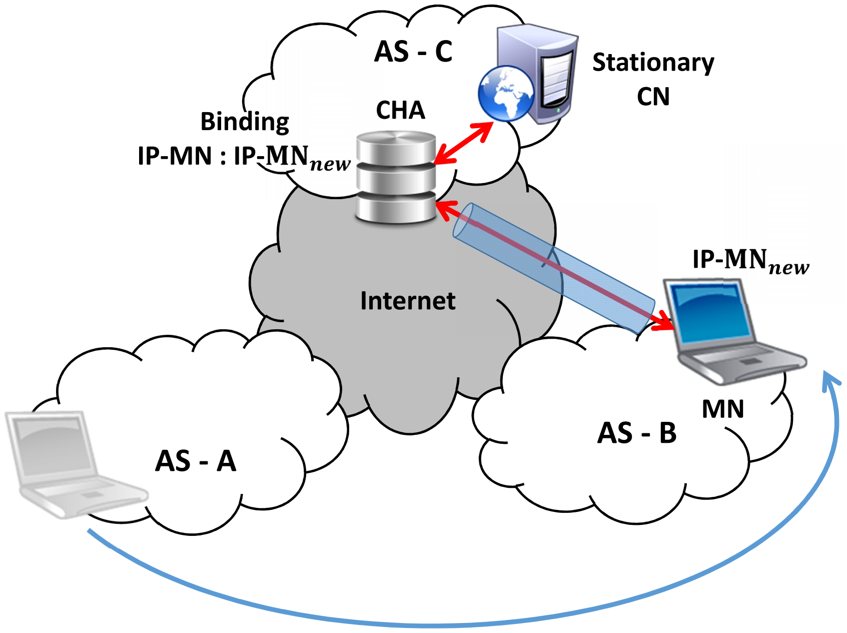


Figure 2.22: Corresponding Network Homing Scheme - Handover.

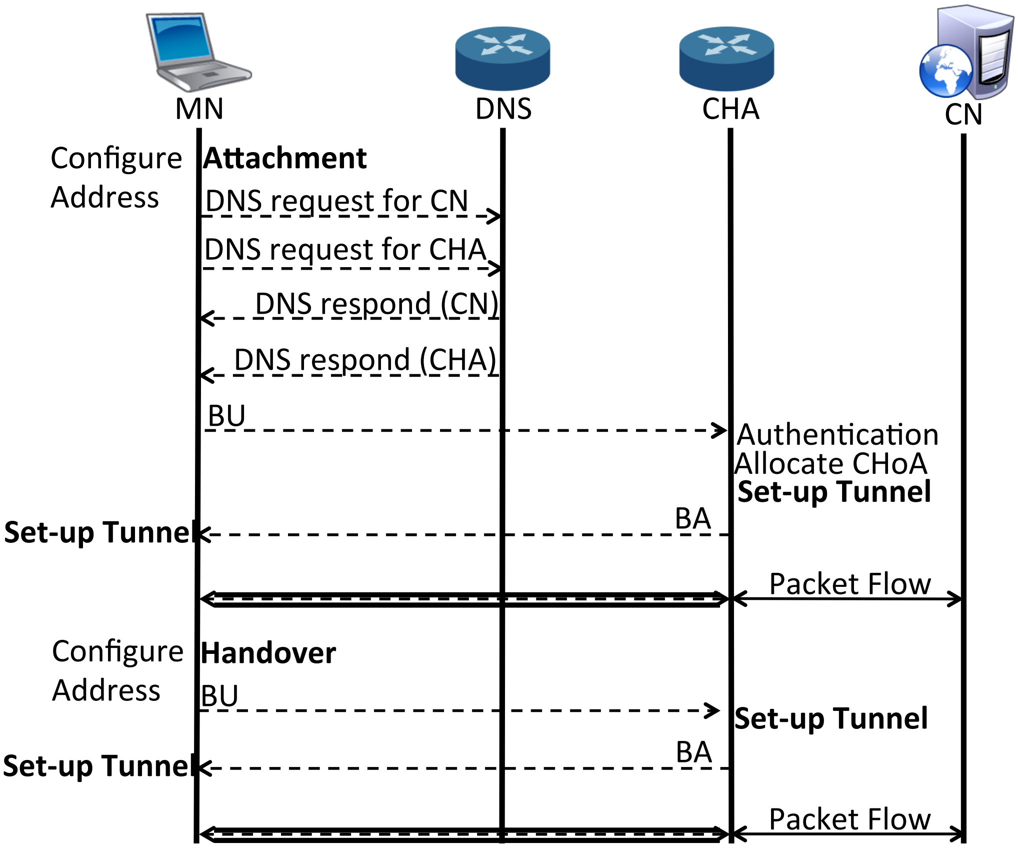


Figure 2.23: Control and Data Flow of Corresponding Network Homing.

After the MN relocates, it detects that it has been assigned a new IPv6 address, similar to the MIPv6. Then, again the MN is required to send a BU message to the CHA notifying its new IPv6 address. This procedure is exactly the same as that of MIPv6. After the CHA is notified about the new IPv6 address of the MN, a new tunnel is established between the new IPv6 address of the MN and the CHA. This facilitates uninterrupted communication between CN and the MN.

2.7 Distributed Mobile IP (DMIP)

All the host-based DMM solutions so far identify the functionality of the MM entity as a unit. However, Distributed MIP (DMIP) [15] uses a different approach than the common practice. It identifies three basic functionalities that exist in MM schemes.

1. Home Address (HoA) assignment
2. Location Management (LM)
3. Mobility Routing (MR)

Rather than duplicating the MM agent as a whole, it distributes these functions in the Internet. For example it argues that the HoA assignment and LM function can be placed at the home network of the MN. Then, the MR function is distributed in the Internet.

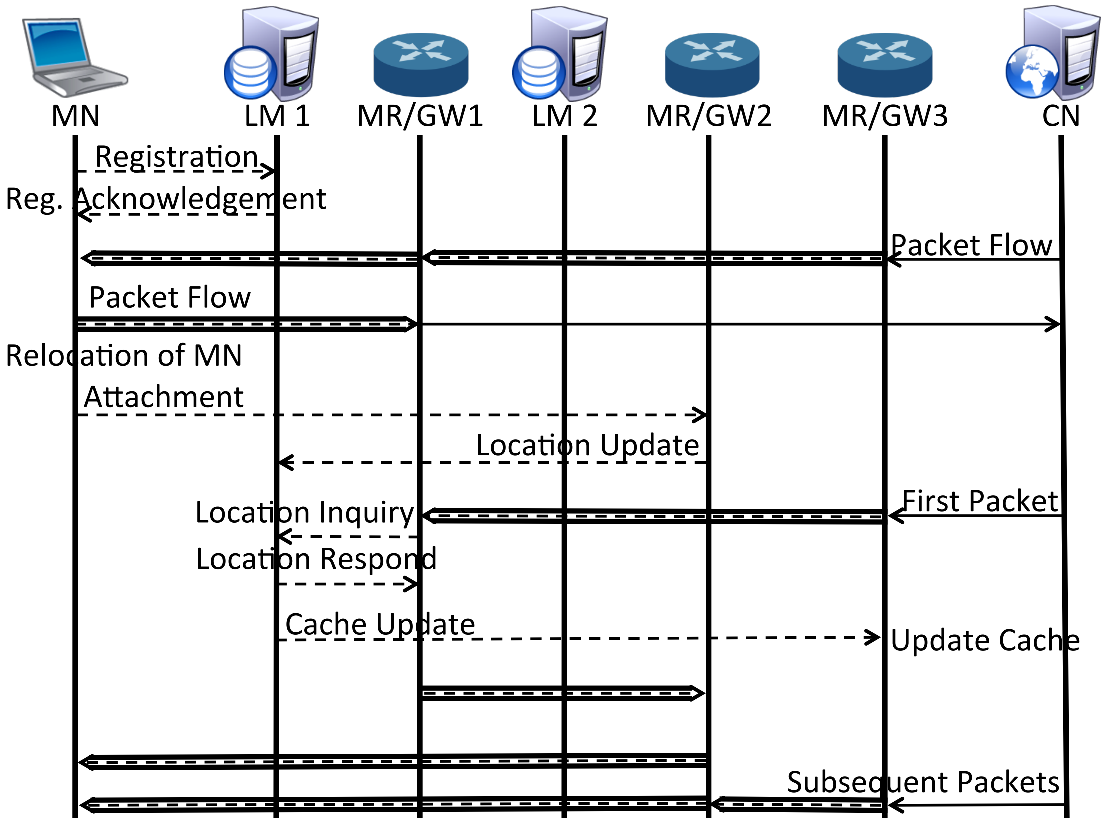


Figure 2.24: Control and Data Flow of Distributed MIPv6 - 1st approach.

Figures 2.24 and 2.25 show the basic outline of DMIP's functionality. First, the MN obtains its IPv6 address identified as HoA from the LM function residing in the home network (Figs. 2.24 and 2.25 (Registration)). After the MN relocates, it will obtain the CoA from the visited network (Figs. 2.24 and 2.25 (Attachment after relocation)). Then, it informs the LM function at the home network of its previously assigned HoA and the current CoA (Figs. 2.24 and 2.25 (Location update)). Triggered by the reception of this information, the LM function at the home network creates and maintains an entry to keep the binding of the HA and CoA. The difference between Figs. 2.24 and 2.25 is the way it handles incoming traffic. Two possible signaling options that can be used when a CN tries to communicate with MN are elaborated.

In the first approach, the MR residing close to the CN always tunnels the traffic which is destined to the MN to the default MR of the MN. Then, if the MN is still attached to the MR, then it will deliver data. (Packet flow of MN before relocation in Fig. 2.24) However, if the MN is no longer attached through the MR, the it inquires the LM function at home. After that, the packets will be tunneled to the new location of the MN. At the same time, it updates the MR which sent the traffic about the binding of the MN. In contrast, the second approach always forces the MR to inquire the LM function of the MN's home network for the location of the MN (Location inquiry in Fig.

Consistency handling of the temporary binding which resides at the MR of CNH is not specified in the paper[15]. Thus, the session continuity is questionable. However, it suggests MN load distribution. Thus, by adopting this method, the amount of space occupied by the MM information can be dramatically reduced.

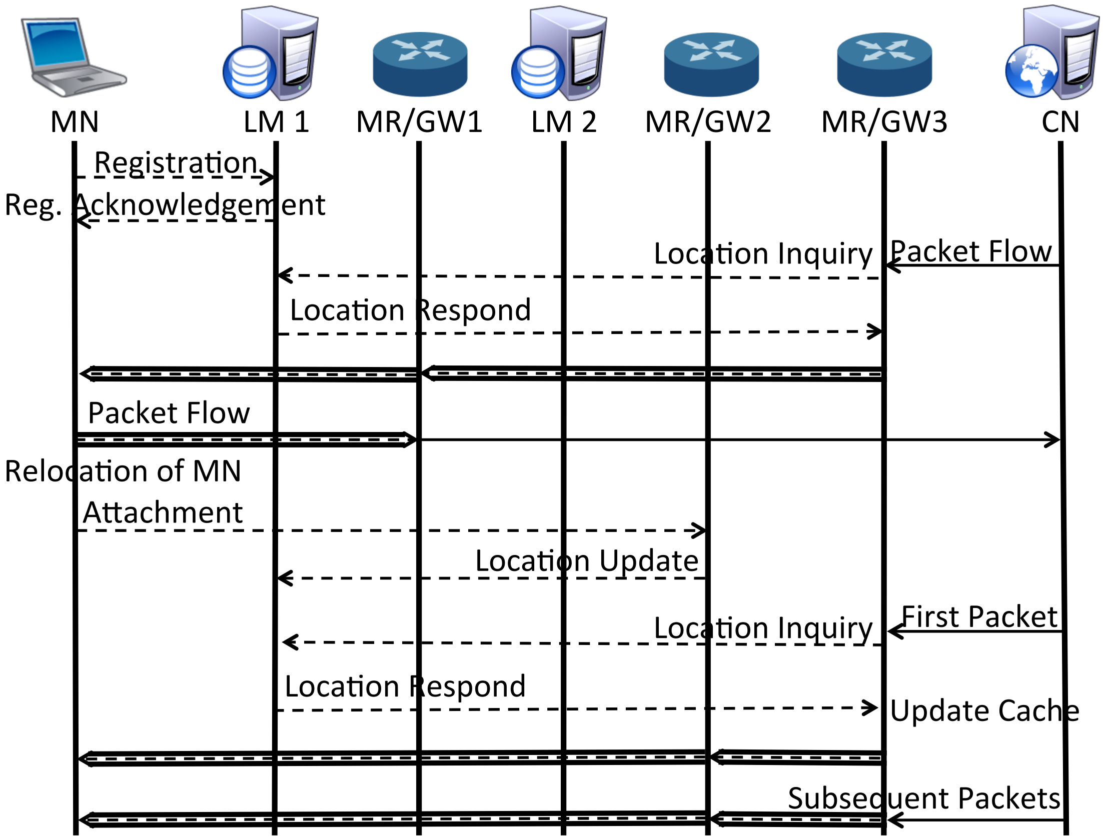


Figure 2.25: Control and Data Flow of Distributed MIPv6 - 2nd approach.

2.8 Proxy Mobile IPv6

In contrast to MIPv6, Proxy MIPv6 (PMIPv6) [25] introduces an approach where network handles the mobility on behalf of the MN. Mobility is transparent to MN. Thus, it does not require special support from MN's protocol stack. This can be considered as one of the initiative attempts which tries to alleviate modifications in the MN's protocol stack in order to support its mobility. Hence, this idea allows legacy MNs to benefit from the MM.

Figure 2.26 shows a general overview of the PMIPv6 during MN initilization. At the place of the HA, PMIPv6 introduces the Local Mobility Anchor (LMA). It is responsible of keeping tracks of the MNs. Since the mobility handling functionality is pushed to the network, PMIPv6 introduces another component to the network. It is known as Mobile Access Gateway (MAG). Similar to the MIPv6, PMIPv6 also defines a CoA. However, the functionality of the CoA is slightly different from the CoA of the MIPv6. This CoA does not directly specify the MN, but it specifies the MAG.

When the MN attaches to the network, it is assigned an IPv6 address as shown in Fig. 2.28. All the MAGs in the network advertise the same network prefix. At the same time, it indicates the same L2 address, such that the MN does not detect the movement. Since the mobility is transparent to the MN, event after a relocation within the local domain, the MN continues to use the same IPv6 address.

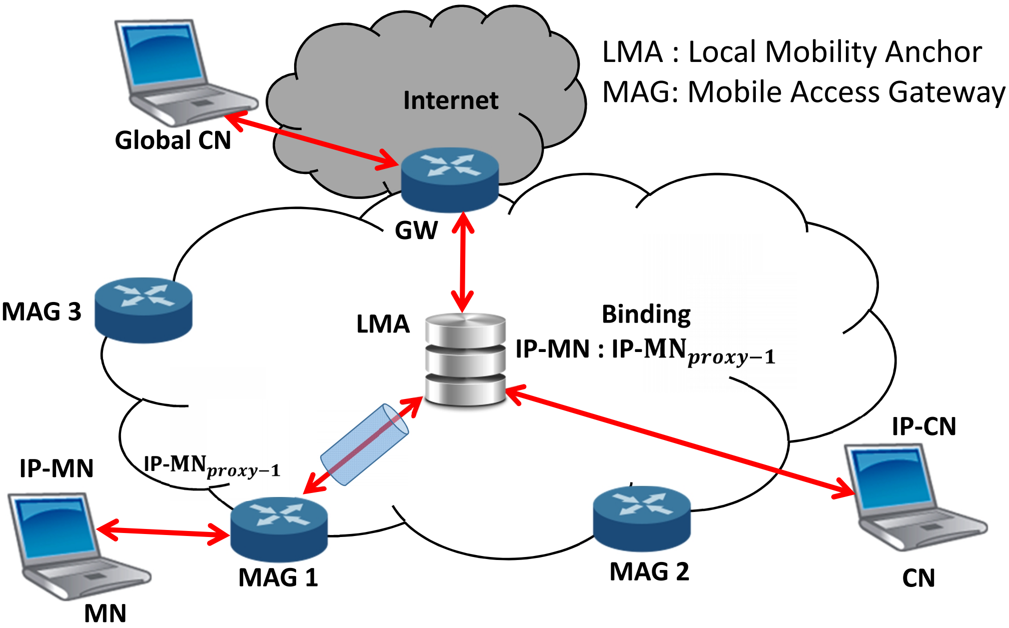

LMA : Local Mobility Anchor
MAG: Mobile Access Gateway
Internet
Global CN
GW
MAG 3
LMA
Binding
IP-MN : IP-MN$_{proxy-1}$
IP-MN$_{proxy-1}$
IP-MN
MN
MAG 1
MAG 2
IP-CN
CN

Figure 2.26: Proxy MIPv6 Architecture - Initial Location.

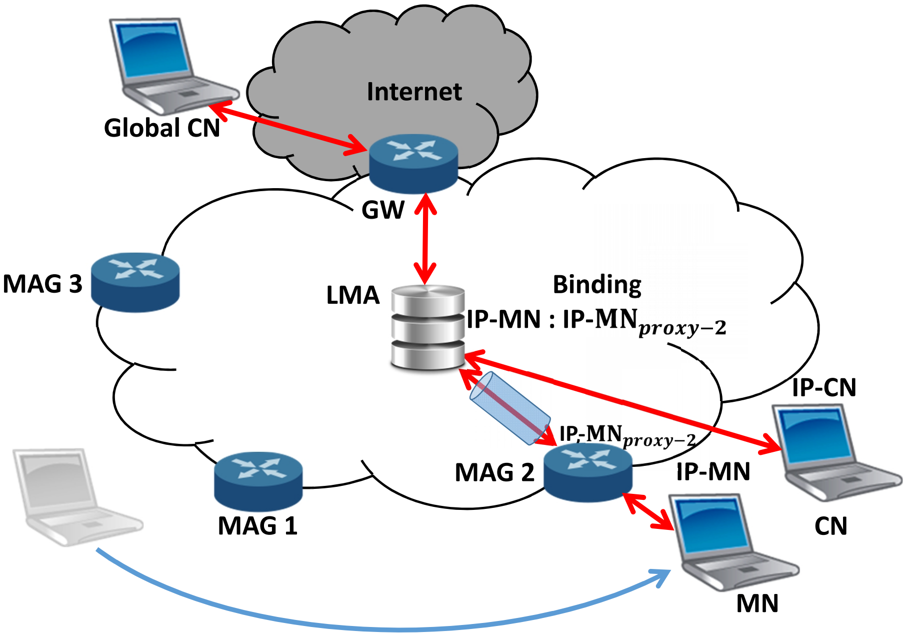


Figure 2.27: Proxy MIPv6 Architecture- Handover.

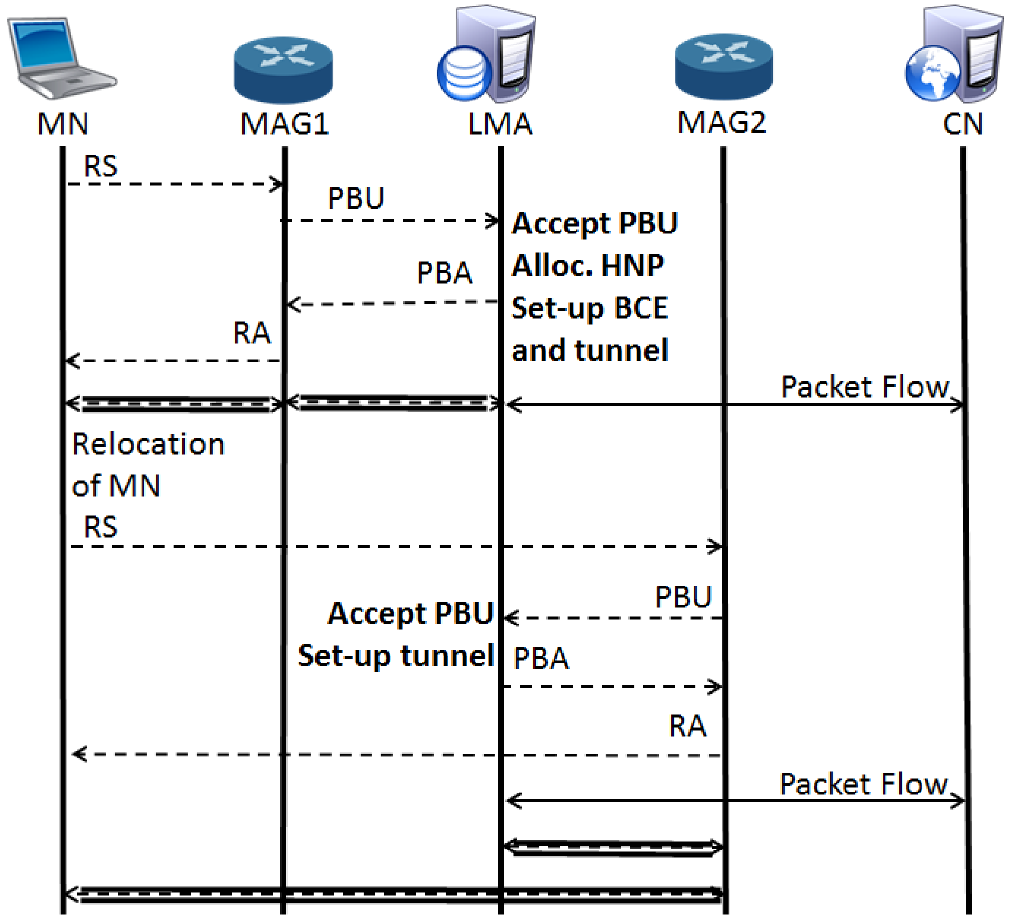


Figure 2.28: Control and Data Flow of Proxy MIPv6.

At the network end, the MAG to which the MN initially attaches (MAG1 according to Fig. 2.28), notifies the LMA of the events on behalf of the MN using Proxy BU (PBU) (Fig. 2.28-PBU). Then the LMA acknowledges by sending Proxy BA (PBA) (Fig. 2.28-PBA). Thereby, the LMA and the MAG create and maintain a bi-directional tunnel in order to exchange traffic of the MN.

When the MN moves to a new location (MAG2 as in Fig. 2.28), MAG2 sends the PBU to the LMA to notify the movement of MN. Then, the binding in the LMA is updated. After sending the PBA, the bi-directional tunnel is established between MAG2 and the LMA, which allows the session continuity and reachable of MN.

2.9 Proxy Mobile IPv6 Based Distributed Anchoring (Extended PMIP)

Network-based MM constantly gained popularity due to its capability to handle mobility transparent to the MN. Thus, with the popularity, it pushes the network to improve network-based mobility handling to facilitate the dramatically expanding subscriber group. PMIPv6 suffers from the burden of having a centralized anchor to anchor all the sessions of MN, which resides comparatively far from the edge of the network.

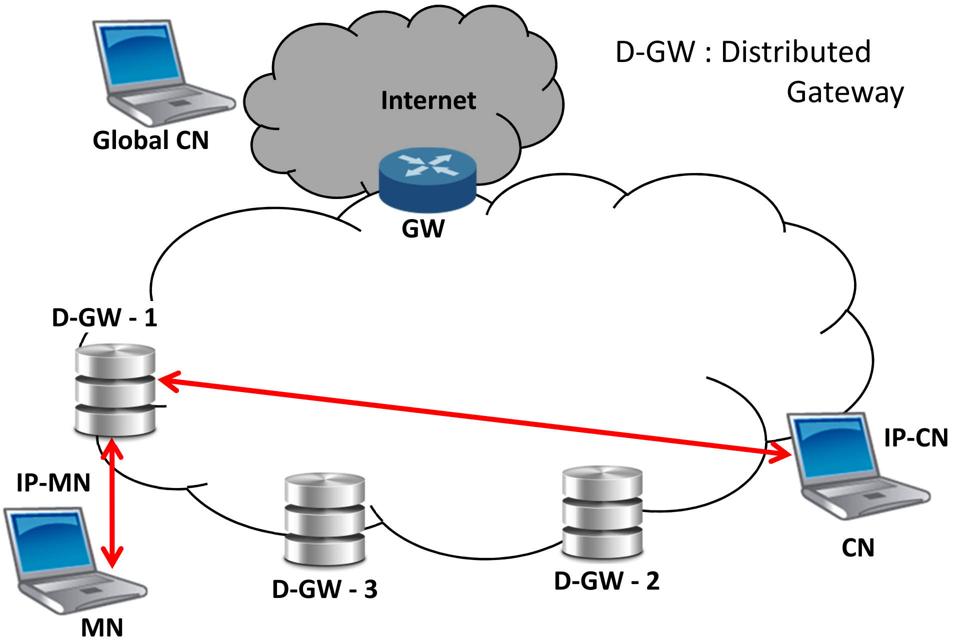


Figure 2.29: Functionality of Extended PMIP - Initial Location.

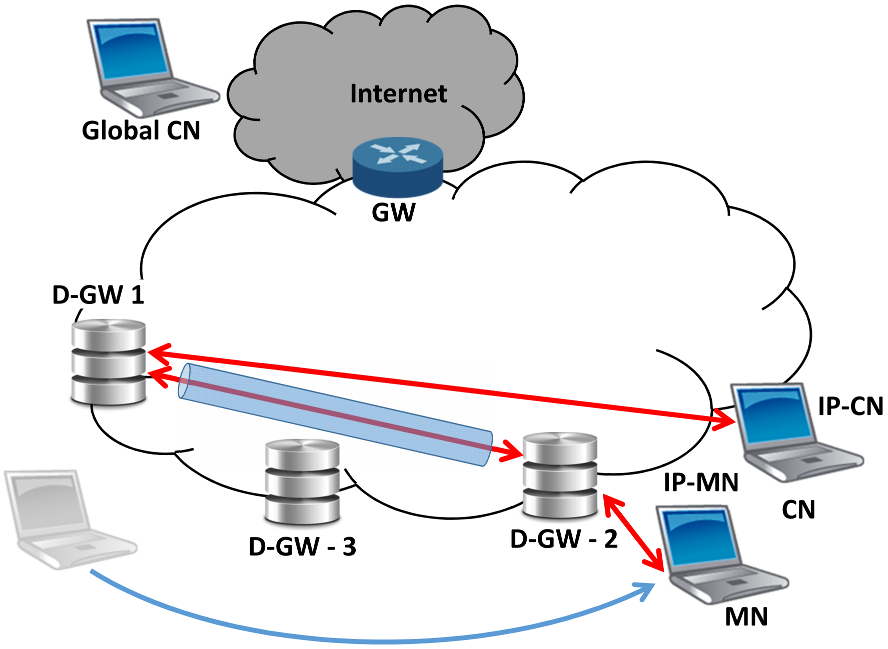


Figure 2.30: Functionality of Extended PMIP - Handover.

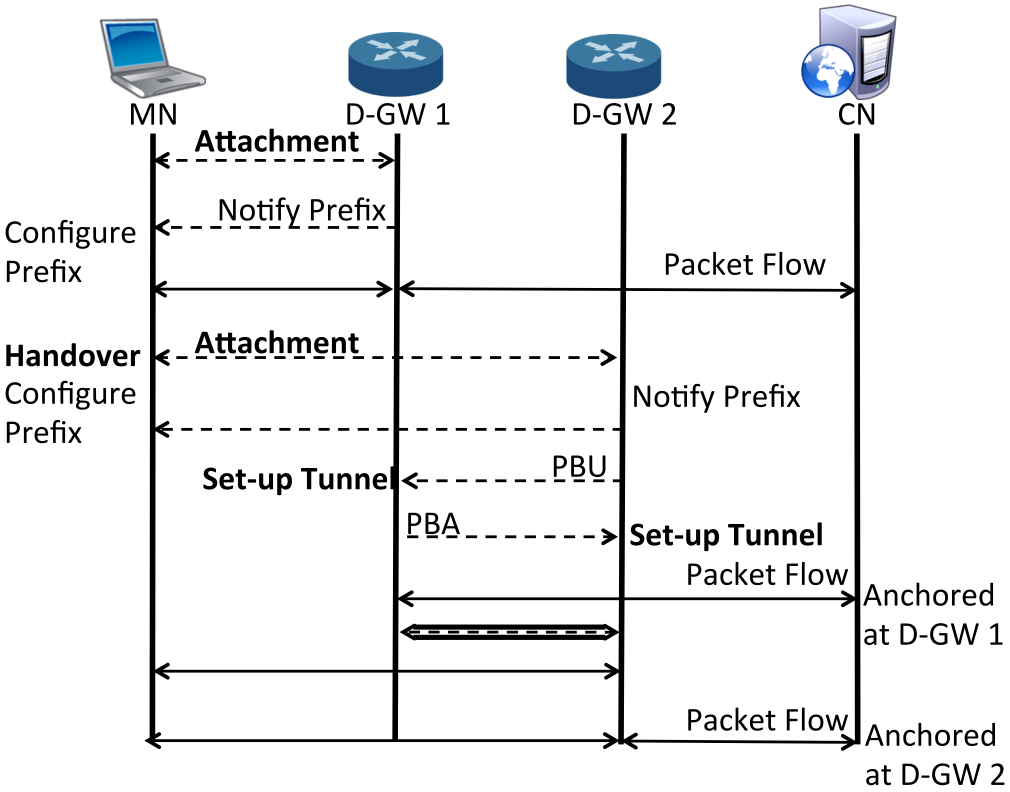


Figure 2.31: Control and Data Flow of the Extended PMIP.

The above mentioned condition introduces packet detour with increased traffic congestion around the anchor. As a solution, Extended PMIP [7] is on of the extensions introduced to PMIPv6 to reduce these issues. It tries to distribute the Anchoring LMA such that the load of MNs in the network can be balanced between the distributed set of LMAs. At the same time, the idea is to push the LMA closer to the edge of the network. Extended PMIP suggest to delegate a static and an unshared address per MN.

Figure 2.31 shows the basic signal and data flows of this scheme. Unlike the PMIPv6, Extended PMIP employs a router called the Distributed Gateway (D-GW), which is enhanced to act as the LMA, the MAG, and the AR. The D-GW is always located at the edge router to which the MNs would connect. MN's current D-GW is known as the serving D-GW (SD-GW).

According to Fig. 2.31, the MN first attaches to D-GW1 and D-GW1 acts as the SD-GW of the MN before it reattaches using a different D-GW. Next, the MN moves to D-GW2. At this point, D-GW2 acts as the SD-GW of the MN and D-GW1 becomes the Anchoring D-GW (AD-GW). The AD-GW is notified by the current SD-GW of the relocation of the MN using the PBU. Once it is received by the AD-GW, it creates a bi-directinal tunnel with the SD-GW. The, AD-GW forwards the packet destined to MN's previously assigned IPv6 address to the new location of the MN. In Fig. 2.31, it is illustrated by the pipe between D-GW1 and D-GW2. Extended PMIP employs a software constructor known as the Distributed Logical Interface (DLIF) at each D-GW. This enables the D-GW to expose itself as multiple routers to the MN, one per active AD-GW associated with the MN. It allows the assignment of multiple IPv6 addresses to the MN. This refrains the usage of the old anchor for the sessions the MN started at

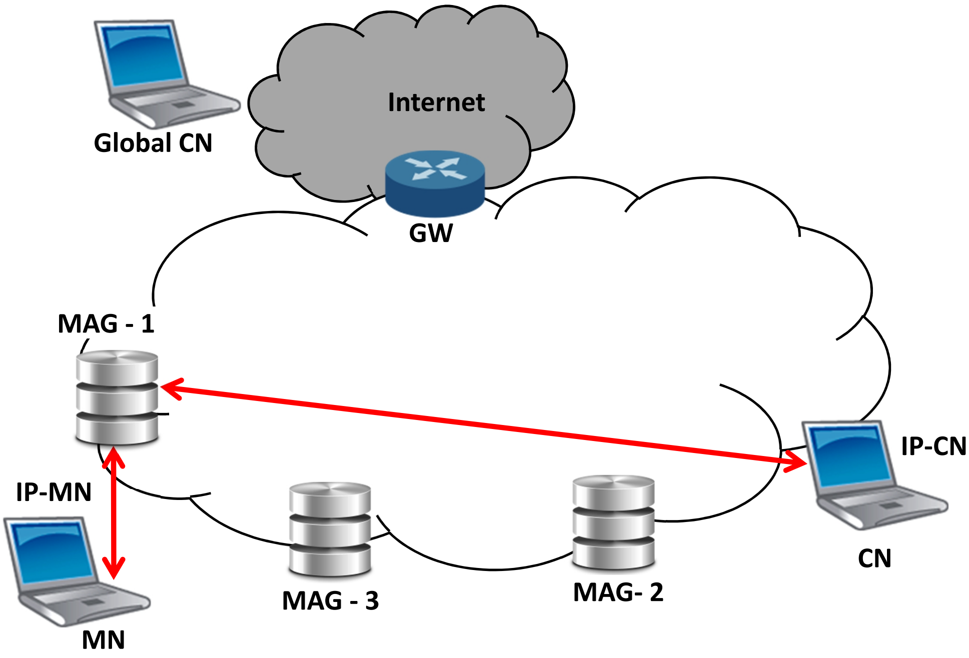


Figure 2.32: Functionality of Distributed PMIP - Initial Location.

the new location. Thus, unnecessary data detours can be avoided. Another advantages of using this scheme is that it allows network specific services which cannot be reached from outside the network to be available anywhere, despite of MN's location.

2.10 PMIPv6-Based Distributed Mobility Management (DPMIP)

Much similar to Extended PMIP described in Sect. 2.9 Distributed PMIP (DPMIP) [36] also tries to enhance PMIPv6. It introduces a fully distributed DMM approach. Figures 2.32 to 2.34 show the architecture of DPMIP and its behavior while the MN resides at the intial location, MN's handover, and at the detection of its detachment from the handover location, respectively.

Figure 2.35 depicts the basic control and data flows of DPMIP. DPMIP suggests to allocate different sub-network prefixes to the MAGs in the local domain. However, they are assumed to advertise the same address prefix. Thus, the mobility can be kept transparent from the MNs. Further, it suggests the usage of DHCPv6 in order to assign IPv6 address to the MNs. Thus, the Router Advertisement (RA) message would always force the MN to use DHCPv6. *DHCPv6 request message* will be sent in respond by the MN. At the initial attachment, MAG1 will intercept this message according to Fig.2.35. On the other hand, if the MN is already assigned an IPv6 address, it will not detect its movement due to the address prefix advertised by the MAG. Thus. this scheme utilizes

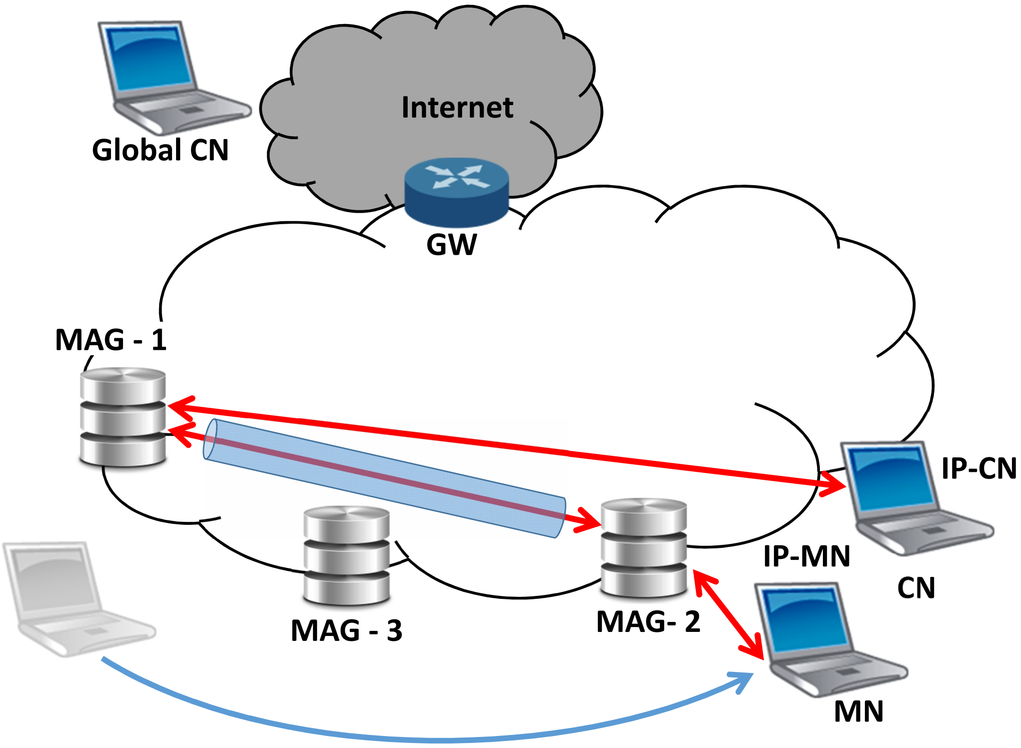


Figure 2.33: Functionality of Distributed PMIP - Handover.

this idea to determine whether it is the initial attachment of the MN or a subsequent relocation.

When the MN relocates, it attaches to MAG2 according to Fig. 2.35. Then, subsequently after inspecting the RA, the MN does not notice the movement. As a result it will continue its communications. Thus, the MAG2 detects the event of the MN's handover by receiving ordinary data packets. Then, MAG2 stores the MN's information into its Binding Cache Entry (BCE) and creates a Binding Update List (BUL) entry, After that, PMIPv6 controlling process of exchanging PBU and PBA takes place between MAG1 and MAG2. At its completion, MAG1 stores the Proxy CoA (PCoA) in its BCE on behalf of the MN. Completion of these tasks allows the MAG1 and MAG2 to establish a tunnel to exchange MN's traffic.

Further, this scheme enhances the performance during the handover by introducing a buffer at the anchoring MAG. If MAG2 detects the detachment of the MN, a Distributed Proxy Binding Release Update (DPBRU) message is sent to MAG1. Following the DPBRU, MAG2 tunnels all the packets that arrived at MAG2 after the detachment of the MN, back to MAG1. When MAG1 receives the DPBRU message, it realizes that the MN is no longer reachable via MAG2. Thus, it stops forwarding the traffic towards MAG2. Subsequently, MAG1 sends a *FLUSH* message to MAG2 to acknowledge the reception of the DPBRU and the end of the data delivery.

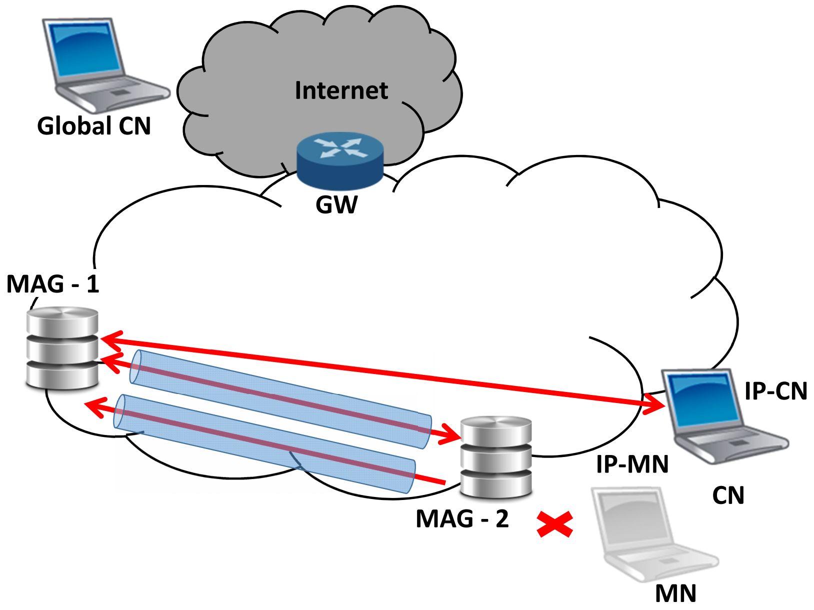


Figure 2.34: Functionality of Distributed PMIP - Detachment from Handover Location.

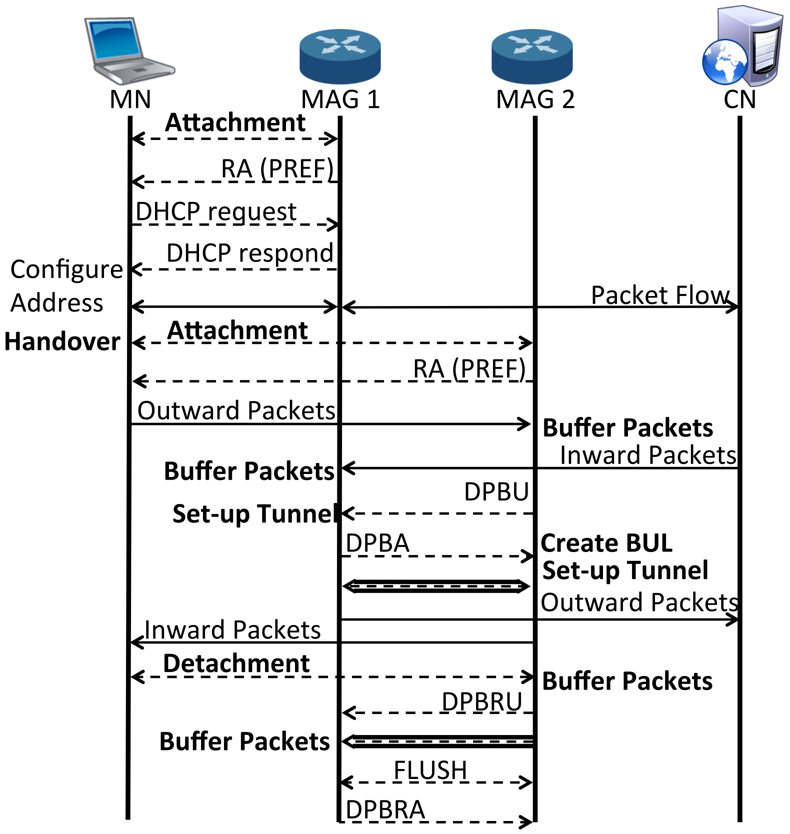


Figure 2.35: Control and Data Flow of the Distributed PMIP.

When MAG2 receives this *FLUSH* message, it detects the end of the traffic that should be tunneled back. Then, MAG2 retransmits it to MAG1. It notifies the end of re-tunneling. After sending the *FLUSH* message, MAG2 removes the BUL entry it maintained for the MN. Finally, MAG2 sends the Distributed Proxy Binding Release Acknowledgment (DPBRA) message to MAG2. This terminates the tunnel.

Similar to Extended PMIP, DPMIP also tries to push the anchor towards the edge of the network. Even though it enhances the data plane performance of the scheme, it introduces a huge employment cost and controlling overhead. That is due to the fact that all the MAGs should maintain mobility information and buffers. Tunneling overhead can also be considered to be high. However, least packet loss can be considered as one of the advantages of this scheme.

2.11 Enhanced Mobility Anchoring (Address Delegation)

Address Delegation [16] can be considered as one of the schemes which focuses on providing optimal routing. It delegates the IPv6 address assigned to the MN, to the new network to which the MN relocates. Thus, rather than simply reserving it to be used as the identifier for MN's older sessions, Address Delegation suggests the IPv6 address to be used as the locator at the same time. General functionality of this scheme is shown in Figs. 2.36 and 2.37. The main control and data flows of this scheme are shown in Fig.2.38.

This scheme lets ARs residing in each sub-network to advertise a defined block of prefixes. When an MN crosses between ARs, existing sessions that can survive an IPv6 address change or the new sessions use the IPv6 address the MN is assigned at the new network. This is known as packet offloading. However, there may exist sessions which cannot survive a change of an IPv6 address. In order to facilitate them, the AR at the new sub-network starts to advertise the IPv6 address which the MN was assigned at the earlier network. This IPv6 address is termed as the *Delegated IPv6 address*.

Relocation scenario in Fig.2.38 shows an instance where the MN crosses between AR1 and AR2. After the MN's relocation at AR2, the network to which AR2 belongs, starts to advertise the *Delegated IPv6 address*. Consequently, AR1's network stops advertising it. Accordingly, all the routing tables in the network will be updated. Thus, packets destined to the *Delegated IP address* will correctly routed to the AR2, rather than towards AR1. However, the convergence time of the routing tables of the network will have an extreme impact on the performance of this scheme.

Even though, ideally, this idea can be implemented easily within a localized domain, the Address Delegation scheme suggests an extension to allow migration of MNs between different domain. I2RS[5] mechanisms or NETCONF[26] are suggested to be used to reconfigure the ARs during localized MM. Address delegation between networks follows the same criterion, assuming that the corresponding gateway routers of the networks run iBGP between them and advertises the aggregate of IP addresses of its sub-networks towards each other. An inter domain routing table updates are suggested to be supported by protocol like IS-IS[66].

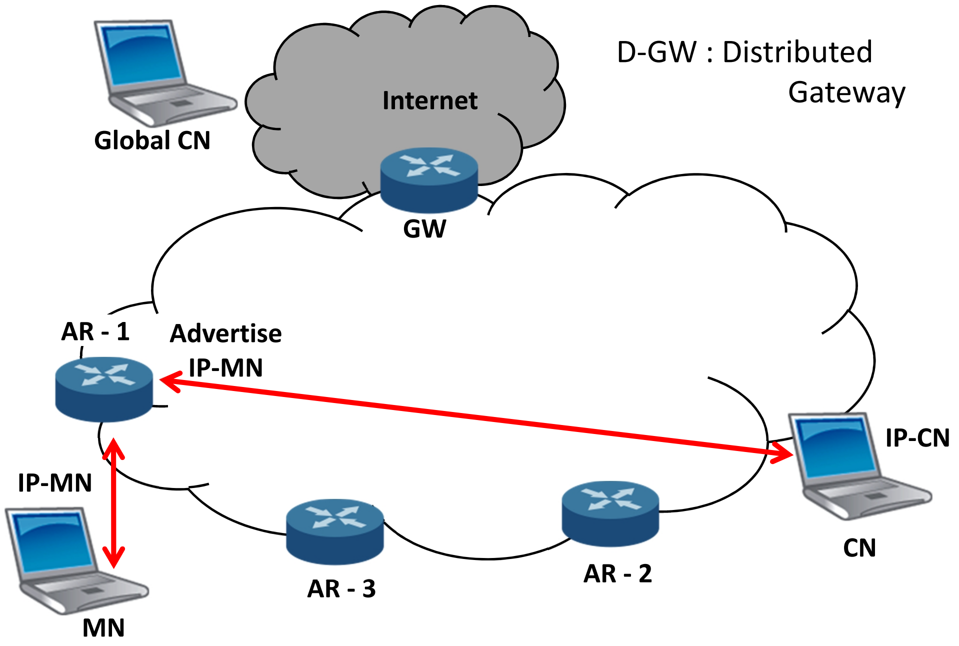


Figure 2.36: Behavior of Address Delegation - Initial Location.

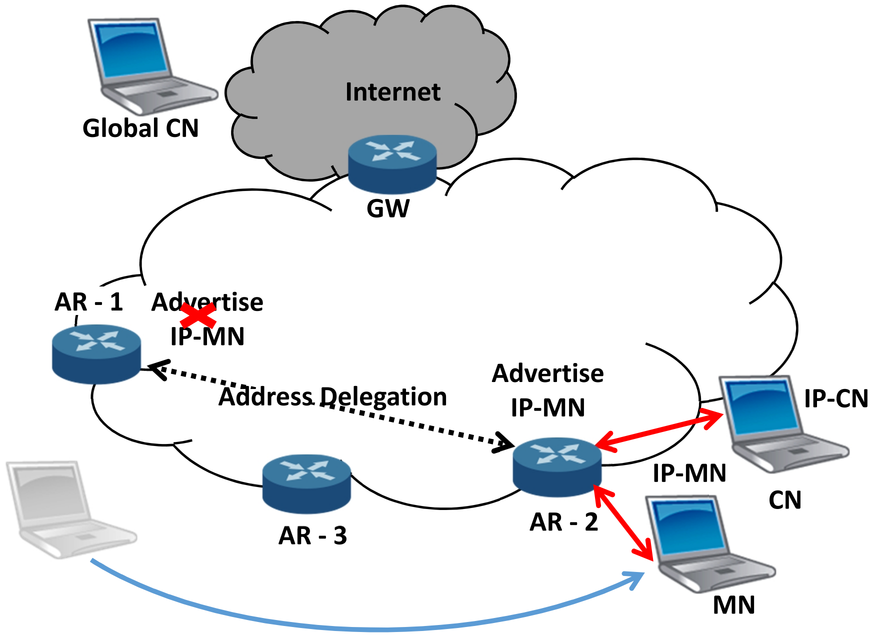


Figure 2.37: Message flow of Address Delegation - Handover.

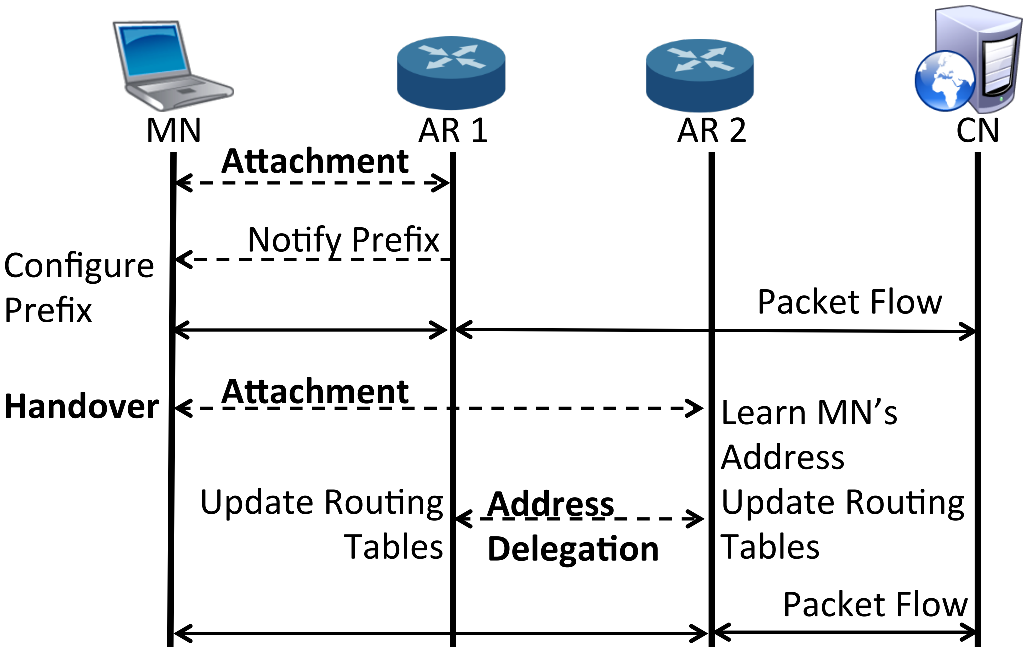


Figure 2.38: Control and Data-Plane Flow of Address Delegation.

2.12 Route Optimization with Software Defined Networking (RO-SDN)

Routing optimization described in in RO-SDN [72] suggest a partially distributed MM scheme based on the Software Defined Networking (SDN) paradigm. SDN can be seen as one of the most trendy paradigms to provide control and data-plane separation. It centralizes the control-plane of the network in a centralized SDN-controller. Further, it removes the decision making components of routers and redefines them to be SDN-switches. These SDN-switches simply fowards packets based on the decisions taught by the SDN-controller. Thus, SDN allows to dynamically set the flows by the means of flow tables in the SDN controller.

The functionality of RO-SDN when the MN is at its default location and after the handover is given in Figs. 2.39 and 2.40. Basic control and data flows of RO-SDN are given in Fig. 2.41. In this scheme, the ARs in the network edge are termed as *Mobility Anchor and AR (MAAR)*. Initially, the MN attaches to the P-MAAR according to Fig.2.41. Then, P-MAAR is set to send the *Packet in* message with MN's ID to the SDN controller for registration. The controller sends the *Packet out* message in responce, including the MN's prefix. At the same time, the SDN-controller stores MN's information in Binding Cache Entry (BCE). In order to enable communication for MN, the SDN-controller sends a *Flow Modify* message to set up the flow table in the P-MAAR in order to setup the data path.

According to Fig. 2.41, the MN relocates at the S-MAAR. At this instance, S-MAAR becomes the service MAAR and the P-MAAR becomes the Previous MAAR. Then, similar to initialization, the S-MAAR sends the *Packet in message* again, including MN's ID, prefix, and the new location. The SDN-controller modifies the BCE.

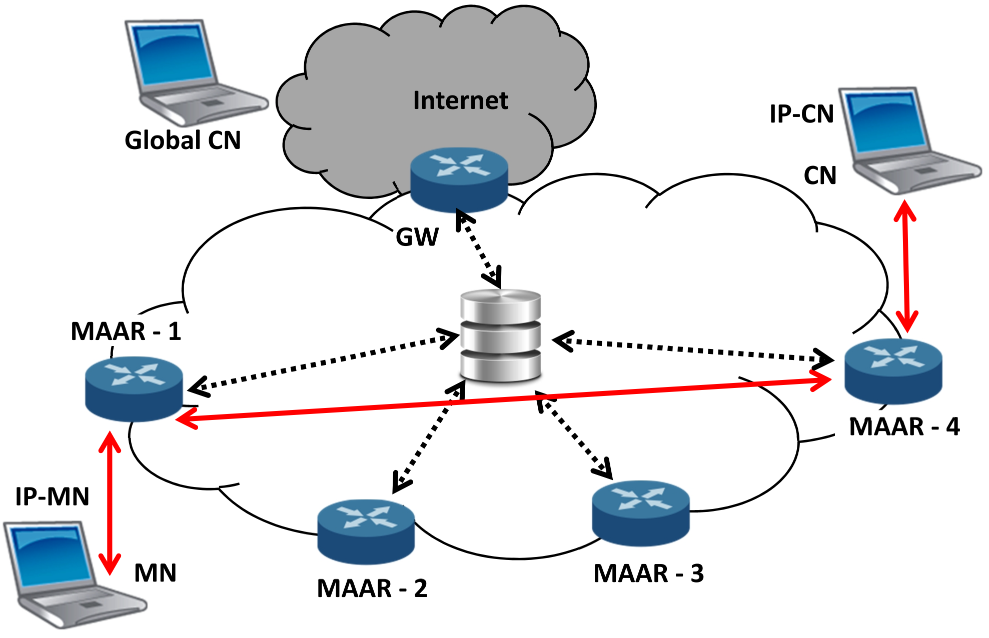


Figure 2.39: Functionality of Routing optimization with SDN - Initial Location.

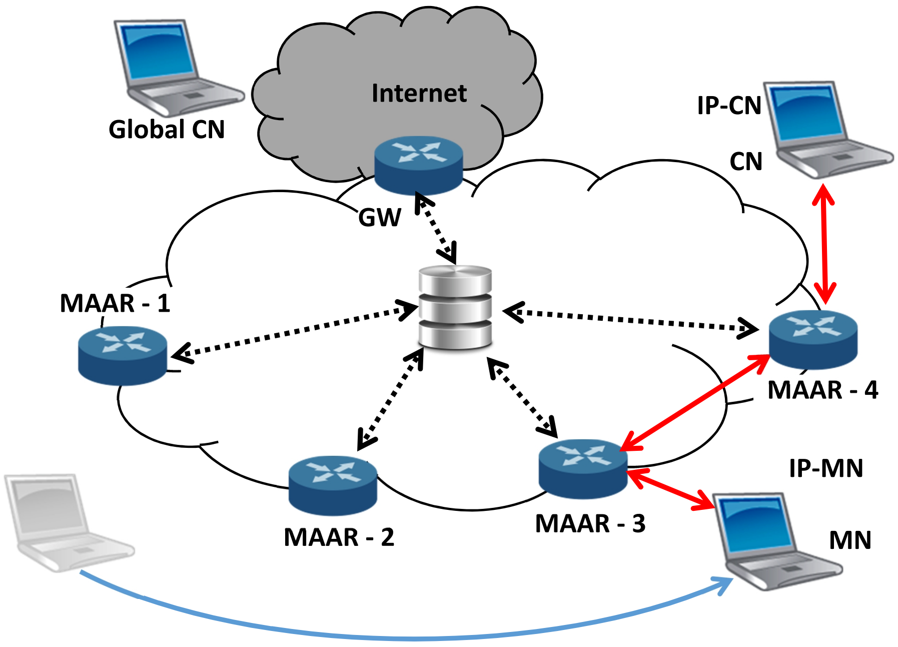


Figure 2.40: Functionality of Routing optimization with SDN - Handover.

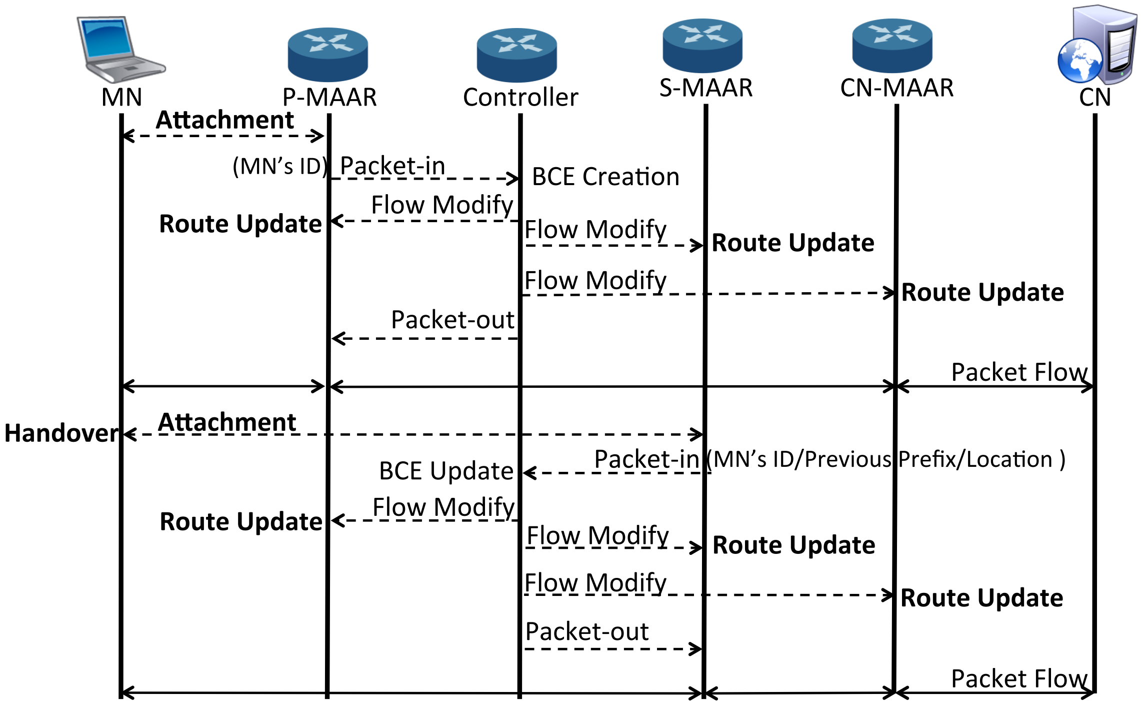


Figure 2.41: Message flow of Routing optimization with SDN.

Now, all the flows of the MN should be fixed to allow the MN to continue its sessions. Thus, the SDN-controller sends off *Flow Modify messages* to the previous and serving MAARs (P-MAAR and S-MAAR according to Fig. 2.41). However, it may require the SDN-controller to modify all the routers which already have flows with the MN.

Similar to the previously discussed Address Delegation, this schemes also aims at attaining routing optimization. Thus, it assures best data-plane performances. However, the convergence time of the network will determine the time it takes to stabilize the network and the performance of the scheme. A single SDN-controller is vulnerable to a single point of failure and attack. Similar to the centralized MM, it also has scalability limitations. On the other hand, if this RO-SDN is employed in a distributed SDN-controller environment, high data redundancy might occur. Further, consistency handling might play a vital role in a distributed environment. It might introduce a large controlling overhead.

2.13 Summary

This chapter introduced different categorizations that are usually seen in the scope of Distributed Mobility Management (DMM). These categorizations are based on the main properties of the proposed schemes. This thesis perceives the categorization of the Mobility Management (MM) scope to be extremely important in determining their behavior. Next, the categorization is detailed and advantages and disadvantages are theoretically identified of each category. A tabulated summary of the selected DMM schemes found in the literature are given based on the main categorizations. Then, the rest of the section provided an overview of those schemes.

Chapter 3

Design and Implementation of SimNetDMM

This chapter introduces the simulator that is constructed to evaluate network layer (L3) DMM schemes, known as SimNetDMM. The design concepts and the implementation are introduced.

3.1 Motivation

Accessing the behavior of the network when MM protocols are employed is indispensable. Especially, it is necessary to confirm their performance when they are employed in real networks. Such networks may include a large number of routers and users. For instance, ISP networks, or the Internet. Thus, creating a test scenario physically is extremely difficult. Simulation can be considered as a prospective approach to analyze protocols in a large network.

Simulators available in the literature, such as the Network Simulator 3 (NS-3) [46], NetSim [44], and OPNET [48] provide least support to simulate non-standard mobility protocols. More to the point, it is extremely challenging to simulate comparatively novel ideas such as DMM. This is due to the lack of support the generic simulators provide towards novel DMM concepts. For instance, control/data-plane split, control-plane distribution and functional distribution of the control-plane least benefit from those simulators. Most recent mobility protocols which adopts DMM are still drafts, which are yet to be implemented. Thus, most of them are not prototyped or tested so far. This has led to an unfavorable condition in the MM research community, where it remains extremely difficult to confirm performances of draft protocols. As a consequence, in spite of having a considerable amount of DMM ideas, standardization seems to lag behind.

3.2 Simulating Mobility Management in the Internet

As already mentioned in Chapt. 1, evaluation plays an important role in networking. It allows the confirmation of the feasibility of a specific scheme. Using generic simulators to evaluate standard mobility protocols with simple topologies is not a difficult task today. That is because simulators like NS-3 [46] comes with MM protocol components. However, the problem that is addressed requires the simulation of draft MM schemes in complex topologies. These drafts includes novel features that are not supported by generic simulators. For instance, as all the standard MM schemes use a centralized MME, general simulators lack features to provide MME distribution and inter-communication of MMEs. These matters should be addressed when creating a MM simulator.

The next major requirement of the novel simulator is recognized as the capability of testing the MM schemes in realistic circumstances. For example, in the Internet or in ISP networks. Simulating such realistic mobility scenarios remains extremely challenging with restricted resources of current simulators.

Basic functionality requirements for a MM simulator can be identified as follows.

1. Collect topology data

2. Simulate packet routing

3. Allow viable MME placement

4. Determine test scenario (Mobility patterns and sessions)

5. Determine Performance Measurements

6. Simulate test scenarios

3.3 Overview of SimNetDMM

Considering the requirements that are identified in Sect. 3.2, a simulator, namely SimNetDMM, is designed focusing on L3 based DMM schemes. The basic functional design of SimNetDMM is given in Fig. 3.1.

Available resources in the literature regarding topologies are expected to be fed to the simulator in order to draw realistic L3 routes. For instance, node layout of a topology, direct connections between those nodes, and properties that can be used to determine the paths that should be traversed en-route to a specific node are considered as candidates. However, they should be carefully determined. Since, the performance of a protocol that is tested will solely depend on the underlying paths between two L3 addresses when L3 simulation is considered. Input files in Fig. 3.1 specifies these prospective data. It is later explored in Sect. 3.4.

Then the basic functionality of the simulator is divided into five main functions. The determination of the sub-functions of these major functions are carried out based on the available outputs. First function is the *Topology Simulator* (Fig. 3.1-(A)). It is expected to build a topology based on available resources and determine IPv6 routing, considering policies that might be employed by network administrator. This is expected to attain accuracy of plotted results.

The second function is the *Test Scenario Generator*, which is depicted by Fig. 3.1-(B). This functionality should be triggered once when a topology is well established. Thus, it is dependent on the first function (Fig. 3.1-(2)). *Test Scenatio Generator* function is expected to generate liable test scenarios. This includes the initial attachment of

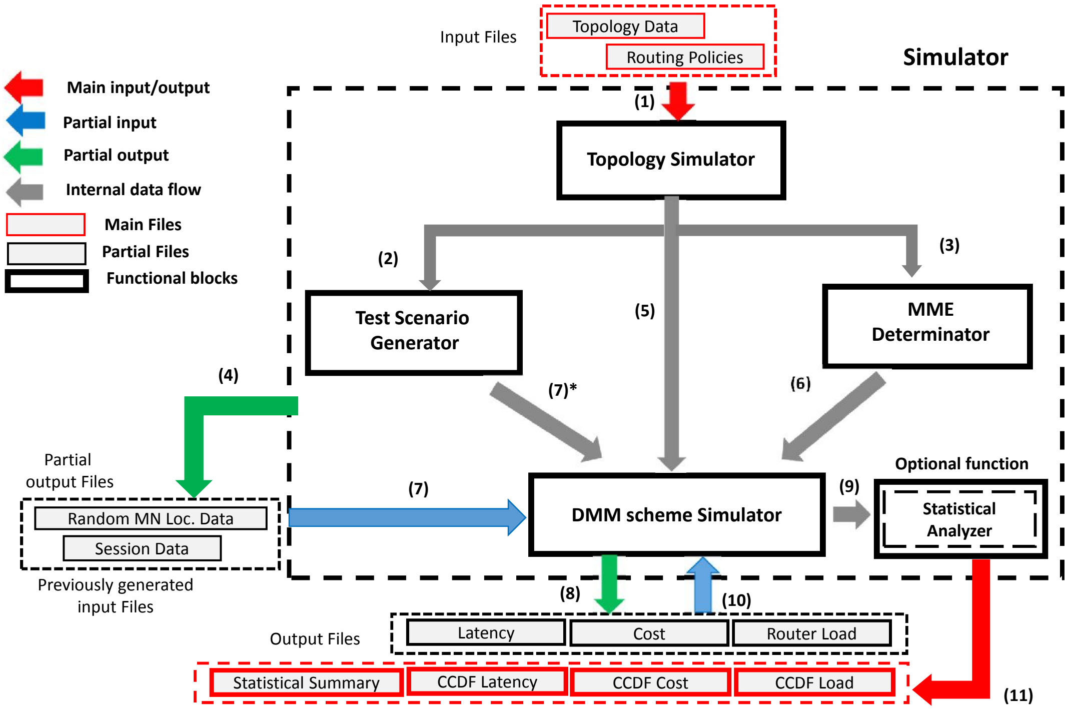


Figure 3.1: Overview of SimNetDMM.

the MN, handover locations of the MN, and the sessions that the MN establishes at each location. In order to determine these, candidate locations like AR, content hosts can be considered. Further, when an AS is considered, it should have facilities for Internet access or content hosting. These properties should be considered when generating a reliable test scenarios.

The third function is *MME Determinator* (Fig. 3.1-(C)). This functionality is also dependent on the topology simulator function (Fig. 3.1-(3)). Once a topology is established and possible shortest paths based on policies are obtained, candidate MME locations should be determined. Considering from the perspective of the network administrator, different aspects like how easily the MME can be accessed, how close the MME resides to the rest of the topology, and how much traffic would flow through the prospective node can be considered as a few of the imaginable concerns. These factors should be taken into consideration. Since the same scenarios are expected to be used over different DMM schemes, it is required to store these MN location and session information as files.

The *DMM Scheme Simulator* (Fig. 3.1-(D)) is the next function of the simulator. This function depends on the three functionalities described earlier (Fig. 3.1 - (5) to (7), and (7)*). It should simulate the control/data-planes of the DMM schemes. By determining the events that occur in L3, determination of performance in terms of user experience, network over head, and overall network performance is expected to take place. Then, the results should be stored for further analyzes.

The fifth and the final function is the *Statistical Analyzer*, as shown in Fig. 3.1-(E). It can be considered as an optional functional block. Analyzing results that are obtained from the *DMM Scheme Simulator* are expected to be summarized. These final results are ultimately used to draw conclusions.

Sections 3.4 to 3.9 describe the methodology to realize the above mentioned functionalities. Then, the implementation and the actual composite functionality of the simulator is given in Sect. 3.11.

3.4 Topology Data Collection Methodology

In order to draw reliable and realistic simulation outcomes, it is vital to have a good set of topology data. As already described in the Sect. 2.1.1, scope of mobility can be local or global. Thus, there are two different types of topologies that can be considered in evaluating MM schemes.

1. Inter-AS topology : (Global) The Internet with a collection of ASes.

2. Intra-AS topology : (Local) An ISP networks which is composed of routers that belong to the same domain.

3.4.1 Inter-AS Topologies

When considering the Internet, it can be perceived as an extremely complicated and a dynamic network. As of its scale and the distribution, it remains abundantly infeasible to sketch its topology. According to the network layer, the Internet can be simplified to a collection of unique administration domains known as Autonomous Systems (ASes) Thus, AS level topology of the Internet can be considered as a simplified version of the Internet, which allows to explore the Internet with a sufficient amount of Information.

There are topology generators such as inet [71], BRITE [11], GT-ITM [14], Igen [31], PRLG [3], NIT [39], RealNet [17], and Tiers [22]. Those are commonly used in reseach purposes. Further, the AS level topology of the Internet is also derived from routing tables collected by RouteViews [56] [57] or Routing Information Service (RIS) [53].

3.4.1.1 Synthetic AS Level Topology of the Internet

The AS level topology generators use different models to model the topology of the Internet. The Waxman model [70], the Erdos-Renyi random graph model [23], and the Transit-Stub model can be considered as a few of the most well-know models. They try to imitate the shape of the AS-level Internet topology following the highlighted properties of the Internet. For instance, the small-world phenomenon [4] can be considered as one of the main properties of the Internet. This phenomenon defines the fact that within the Internet, each node can be approached by another within a defined small amount of hops.

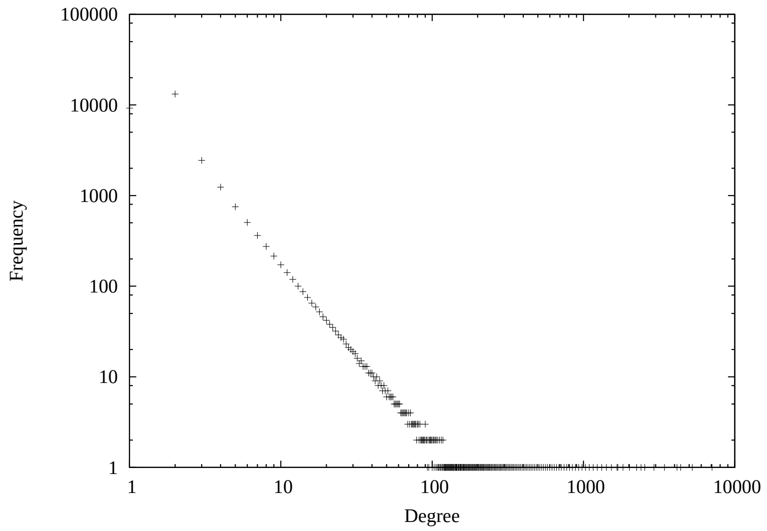


Figure 3.2: Degree Distribution of the inet Topology.

Among the available topology generators, the inet topology generator is selected, based on the fact that it generates a topology which closely resembles the real Internet. This is justified in [71] using the following measurements. It states that the out-degree verses the rank power law graphs of the Internet and the inet generated topology are almost identical. This backs the idea that the inet topology generator imitates it fairly. Further, the resilience and the pair size within h hops of the real Internet and the inet generated topology is also shown to be closely similar. Thus, the expansion factor of the Internet can be seen to be imitaed by the inet topology generator. Very similar normalized Laplacian spectrum of a graph justifies the selection of the inet topology generator well. Finally, the average path length, or in other words, the characteristic path length, average eccentricity and distortion also reflect similarities. However, there are few features which do not show good correspondence. The maximum clique size and clustering coefficient characteristics do not agree with the exact nature of the Internet.

For the purpose of simulating MM schemes in the Internet, a topology consisting of 30,000 AS nodes is generated using the inet topology generator. The degree distribution and the complementary cumulative degree distribution of the generated topology are shown in Figs. 3.2 and 3.3, respectively. Further, the all pair minimum cost distribution of the inet generated topology is depicted in Fig. 3.4

3.4.1.2 Mapped AS Level Topology of the Internet

Center for Applied Internet Data Analysis (CAIDA) [12] is one of the well-known collaborative efforts among organizations that contribute to the composition of the Internet. Those organizations belong to commercial, government, and research sectors. CAIDA focuses on engineering and maintenance aspects of the Internet which requires greater collaboration between different sectors mentioned above.

CAIDA maintains a greater variety of data regarding the Internet. For instance, anonymized Internet traces summary statistics, IPv4 / IPv6 AS Links, AS Ranks, and AS relationships. They are only a few of many data categories CAIDA publishes.

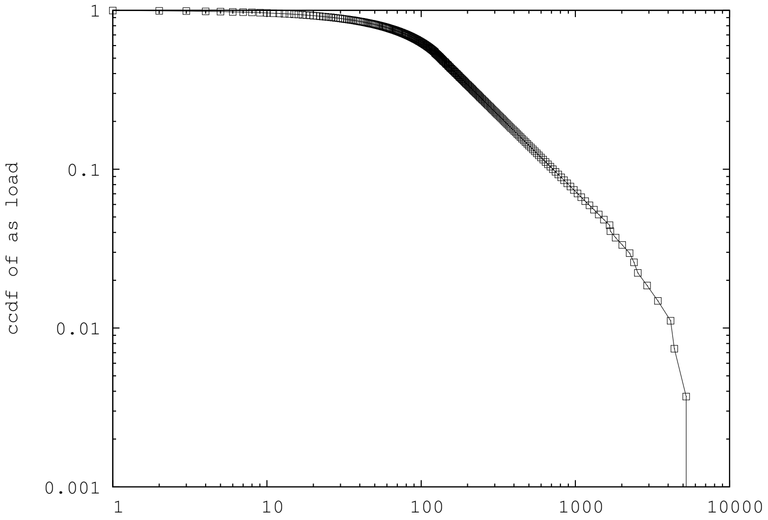


Figure 3.3: Complementary Cumulative Degree Distribution of the inet Topology.

Table 3.1: Percentage of Nodes Having Links.

No. of Links	0 links	1-100 links	more than 100 links
Peer-to-Peer	89%	10.5%	0.5%
Customer-to-provider	0.3%	99.1%	0.6%

When the mobility handling is considered, It is extremely vital to have an accurate understanding about the collaboration of ASes. Thus, by understanding the relationships of ASes, the real data flow within the Internet can be determined.

In order to check the efficiency of MM schemes when they are employed in the Internet, the AS ranks and AS relationships data published by CAIDA are used [13]. The selected data set contains the inferred relationships of 30,742 AS nodes. The total number of the observed links between the AS nodes sums up to 86,711. These relationships are categorized into two types.

- Customer-to-provider relationships
- Peer-to-peer relationships

An example scenario is shown in Fig. 3.5. *Customer-to-provider* relationship specifies that the customer AS pays the provider AS to get the data forwarding facility. In contrast, *peer-to-peer* relationship can only be beneficial to the peer AS itself and the sub-AS nodes or the customer AS nodes of the peer AS nodes. This link is based on the barter traffic exchange between ASes. It implies that this link cannot be used to transit data traffic originated from other AS nodes except for the ones mentioned above. There are 28,368 peer-to-peer links and 58,343 customer-provider links in selected topology. Table 3.1 gives an overview of the AS relationships observed in the used data set.

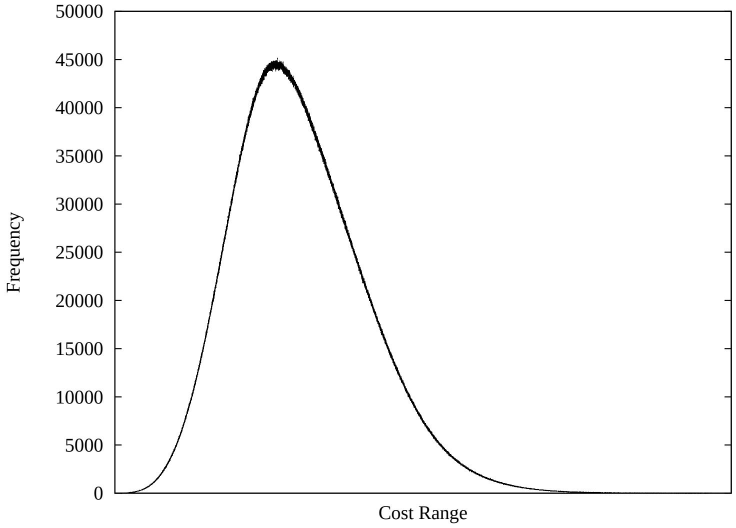


Figure 3.4: All Pair Minimum Cost Distribution of the inet Topology.

3.4.2 Intra-AS Level Topology of the Internet

In contrast to the AS level topology, internal topology of the ISP networks looks at the routers and their correspondence in the network. ISP networks prefer to keep their AS network topology hidden as already mentioned above. Thus, real topology data is not available for research purposes. Alternatively, modeling techniques are used to model typical topologies when required in research purposes.

BRITE[11] and BA[6] are example mathematical modeling techniques which are based on the power law[24] degree distribution. Another approach is mapping ISP topologies using publicly available traceroute information[8][65] which is capable of tracking router hops and transit delays encountered by packets en-route to various destinations.

However, selecting one generic AS topology is extremely difficult. It is because, the ISPs are extremely heterogeneous. In general, they can be classified into three major groups. Thus, in order to understand performance of each scheme in different environments , ISP classification based study is considered.

The literature consists of such work which classify ISP networks into different groups based on their well-know properties.

One such work[22] tries to characterize the Internet hierarchy based on different viewpoints. It discusses a technique which divides the Internet hierarchy into levels known as *tiers* based on the AS relationships. It also takes into consider the degree of the nodes to understand the importance of the nodes. It identifies three major *tiers*. A list of ISPs categorized into three tiers (tier 1, 2, and 3) based on the above classification[64] are used as the reference for this thesis.

Tier-1 consists of the ISPs that reside in the core of the Internet. Thus, they are expected to be larger in size having more routers and links. Further, they tend to cover multiple continents with higher AS degree, maintaining higher number of peerings with other ISPs. In overall, they mainly facilitate the rest of the ISPs with interconnections. There are 22 *tier-1* ISPs as in 2002[64].

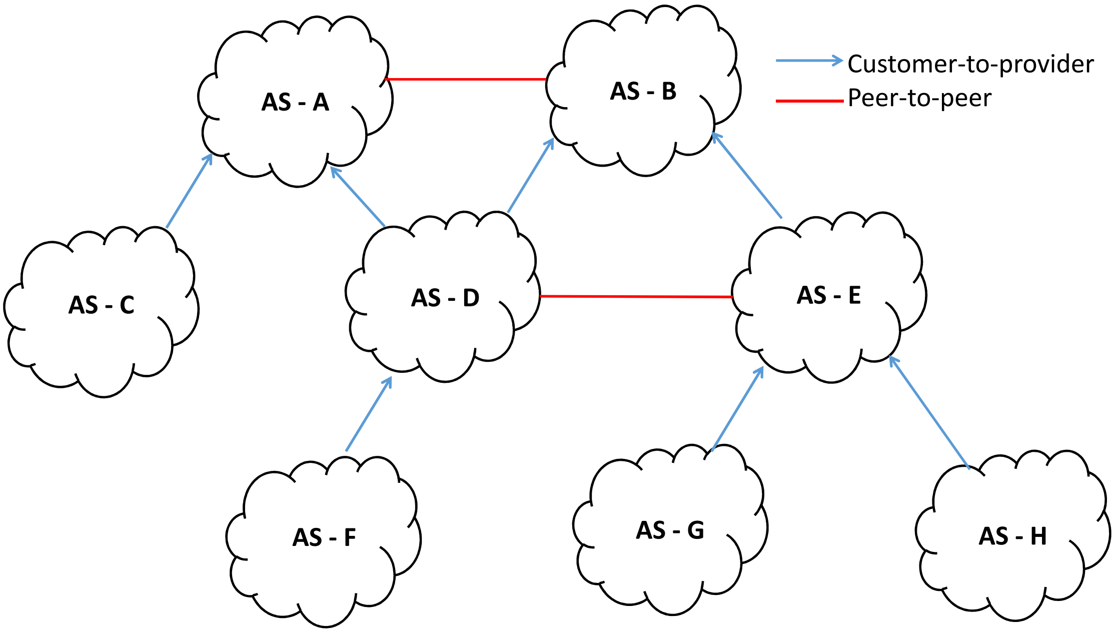


Figure 3.5: Inter-AS Relationships.

Tier-2 consists of the ISPs which are relatively smaller in degree as well as in size with compared to *tier-1* ISPs. They are majorly assumed to be engaged in transiting.

The final tier, *tier-3*, contains the ISPs which solely purchase access to the Internet from *tier-2* or *tier-1* ISPs. These can be considered as leaf ISPs with least degree.

3.4.2.1 Synthetic Internet Service Provider Topologies

In mathematically modeling the ISP topologies, a well-known set of models are frequently used. For instance, Waxman model[70] and Barabasi-Albert (BA) model[6] can be considered as two of the most common models.

Considering the fact that it is a well known topology generator used by a majority of researchers, the BRITE[11] topology generator is used to generate synthetic ISP topologies for the evaluation. BRITE topology generator uses recognized models such as Waxman model[70] and Barabasi-Albert (BA) model[6] to generate viable topologies.

Three different router-level topologies are generated as representative ISP topologies, utilizing Barabasi-Albert model considering random node placement following an incremental topology growth. The *x,y* coordinate information of the nodes generated by the topology generator is used to locate the nodes. Basic information of the topologies are shown in Table 3.2.

The first router-level topology is of a small ISP, which is synthetically generated using the BRITE topology generator, utilizing the above described parameters with 250 router nodes. The generated topology has 497 interconnections between routers which

Table 3.2: Synthetic Intra-AS Topologies.

	Topology	Number of Nodes	Number of Edges	Maximum Degree
1	Small	250	497	53
2	Medium	750	1497	91
3	Large	1500	2997	96

are termed as *edges*. The number of edges per router is known as the *degree*. As the BRITE topology generator does not assume duplex edges, the minimum degree possible is set to 2, such that a duplex connection is established using a pair of incoming and outgoing edges. Maximum (two-way) degree observed is 53. The second test topology is considered a moderate topology, having 750 routers and 1,497 edges, whereas the maximum degree is 91. The final topology is a comparatively larger topology with 1,500 routers, which contains 2,997 edges. The router with the maximum degree has a 96 two-way edges.

3.4.2.2 Mapped Internet Service Provider Topologies

Rocketfuel[54][63] is one of the efforts which map various ISP network topologies at the router-level.

To make sure that performance evaluations are consistent with the real ISP topologies, example topologies from each tier are selected as referential topologies. One high degree topology which is close to the Internet core (Sprint) is selected from *tier-1* . Then, two smaller ISP networks residing significantly farther from the core (Telstra and Exodus) are selected from *tier-3*. This is due to the fact that leaf topologies tend to have the highest variety amoung the three tiers. Finally, one moderate ISP (Tiscali) from the *tier-2* is also selected.

The topology data of these topologies are obtained from Rocketfuel [63] repository. The geolocation maps of the backbone topologies of ISP networks are shown in Figs. 3.6 to 3.9 (source: Rocketfuel Maps[55]).

Apart from the routers and the router inter-relationships (edges), Rocketfuel also hosts data about the roles played by the routers and the geolocations of the routers. ISPs in the Internet are identified by a unique Autonomous System Number (ASN). The number of neighboring ISPs of the concerned ISP is defined as the Degree of the ISP where it plays an important role in understanding the connectivity an ISP has with the rest of the Internet.

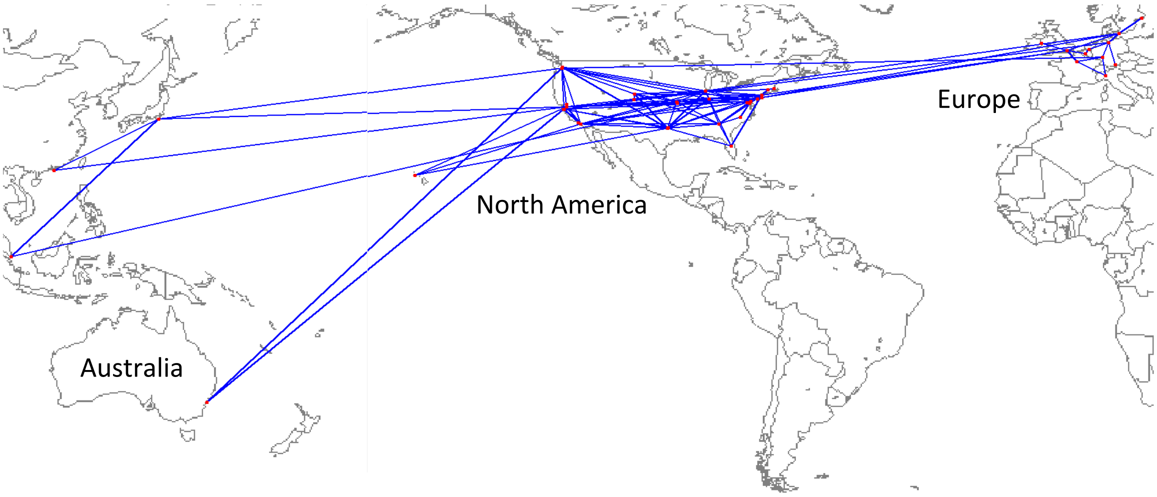


Figure 3.6: Sprint : USA (ASN: 1239) (source: Rocketfuel Maps[55]).

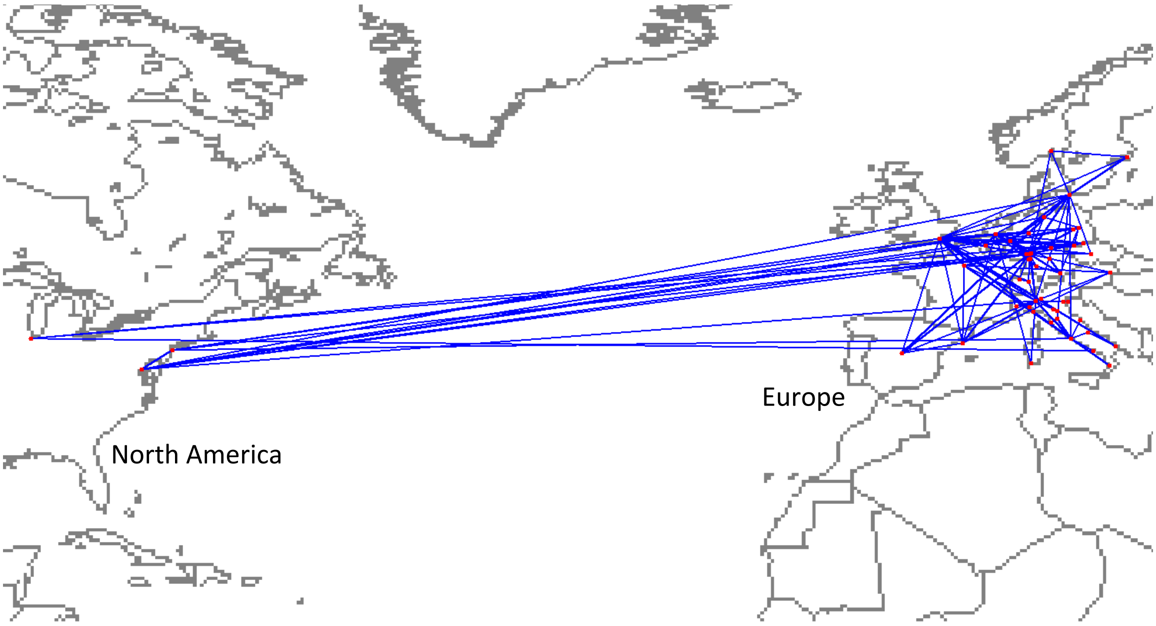


Figure 3.7: Tiscali : Europe (ASN: 3257) (source: Rocketfuel Maps[55]).

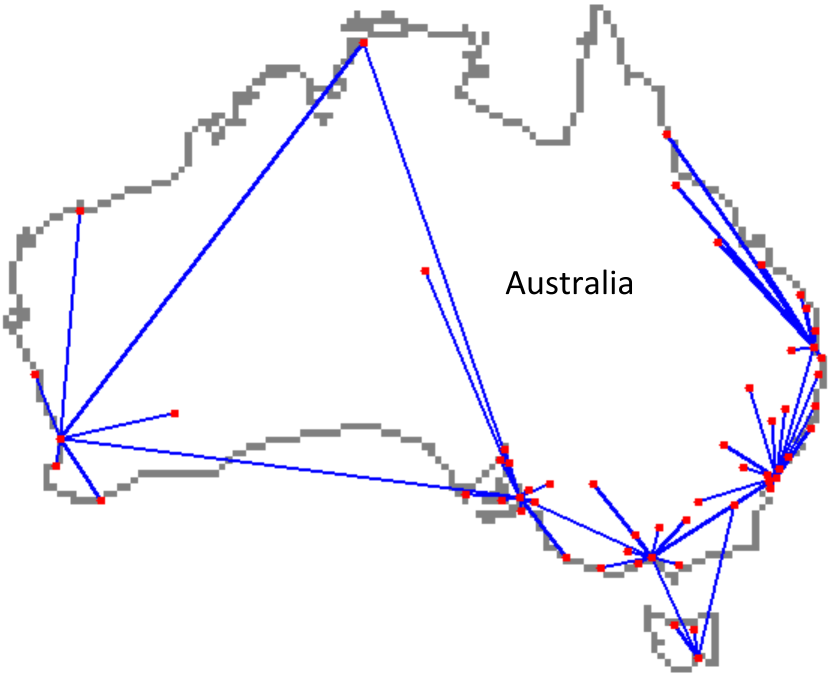


Figure 3.8: Telstra : Australia (ASN: 1221) (source: Rocketfuel Maps[55]).

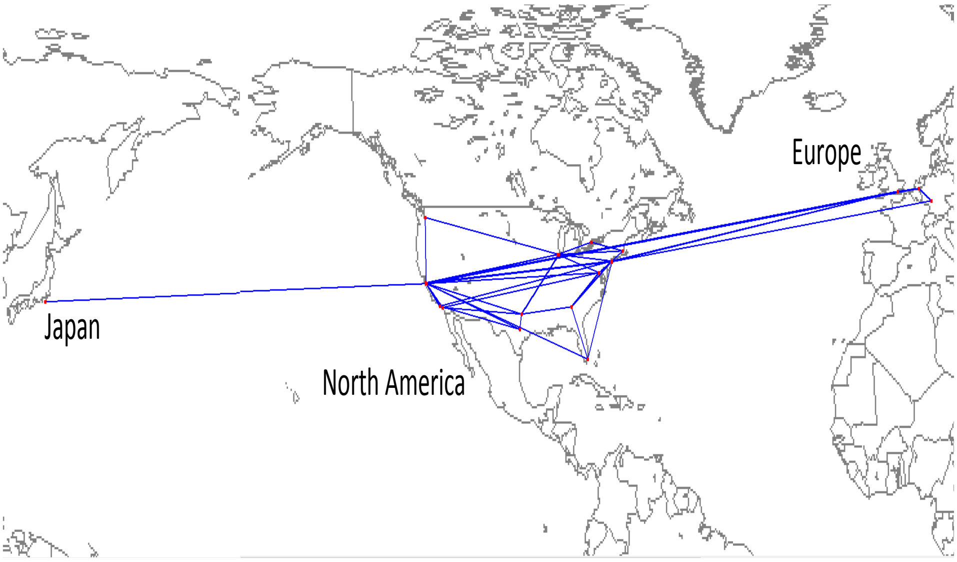


Figure 3.9: Exodus : USA (ASN: 3967) (source: Rocketfuel Maps[55]).

ISPs used in the evaluation are basically considered to be composed of two major portions. One is the backbone, which is a part of the network infrastructure that is connected by higher bandwidth links. The other portion is the Point of Presence (PoP). The roles of routers can be categorized into 4 groups, whereas they reside in the above portions according to the role they play. The roles played by routers are,

1. Backbone routers (BBs)

2. Access Routers (ARs)

Table 3.3: Dimensional Information of Selected ISP Network Topologies.

AS Name	Tier	AS Deg.	BB RTRs	BB Links	PoPs
Sprint(1239)	1	1,735	315	1,944	43
Tiscali(3257)	2	326	161	656	50
Tesltra(1221)	3	66	108	306	61
Exodus(3967)	3	43	79	294	23

3. Peering Points (PPs)

4. other / Undetermined

Backbone of the network is composed of BBs, whereas the PoP includes network elements such as the ARs and the PPs. ARs are supposed to provide Internet connectivity to individual users. The gateways or PPs connect the ISP with external ISPs. If two ISPs have multiple number of PPs, it requires special determination process to determine the PP to be used at a given instance. Basic information about the selected set of ISPs is provided in Table 3.3.

Geolocation of the routers also can be considered as one of the important properties of routers. It might determine the extensions to be imposed on the router. For example, if the router has high connectivity, then it can easily influence the other routers. Thus, hosting alterations or updates from that point will let the updates to reach alot of nodes quickly. Thus, it can be considered as a candidate location to host some service. Geolocation based mapped backbone topologies of the above-mentioned ISPs are depicted in Figs.3.6-3.9.

3.5 Packet Routing

Packet routing in the Internet is a complex task. That is because, the packets might have to travel through portions of the Internet that belong to different administration *domains*. In general, ISP / AS domains maintain their topology information unexposed. This is due to the security considerations. Exposing the topology can make it vulnerable to hackers. Thus, instead of exposing the topology, domains tend to expose only the IPv4/IPv6 address ranges they host, and the domains that can be reached via these domains. Sometimes, based on the negotiations between domains, some domains may expose more information to the other domains. Based on the locations of the source and the destination, routing can be considered to be of two types.

- Intra-domain Routing
  Routing traffic originated from an ISP domain to a destination which resides in the same ISP domain is considered as Inter-domain routing. In this situation, the packets would never leave the ISP network. Delivering incoming traffic originated from an external ISP network, which is destined to a local node also belongs to the same category. Since, once the packets arrive at the gateway, after that it can

be considered as an internal packet delivery scenario. This is because they will not leave the network once after they enter the ISP domain.

- Inter-domain routing
  When the destination resides outside the ISP for locally originated traffic or in case of a transit, it requires the traffic to exit the ISP network via an appropriate gateway. In order to route such packets, it requires the neighboring ISPs to collaborate between each other. This routing scenario is known as inter-domain routing.

The collaboration taken place can be further divided into two. This categorization is based on the immediate neighboring ISP's role in routing the packets.

- Peering :
  When the packets are destined node which belongs to the neighbor ISP.

- Transit :
  If the packets are only taking a transit in the neighbor ISP, and the destination does not reside in neighbor ISP.

In order to obtain legitimate simulation results, it is vital to simulate the real routing approaches adopted by each ISP. Thus, it is required to identify and construct the criteria that determine the routing paths that are taken in each situation. *Routing policies* determine which of the available links to be taken in intra-doamin and inter-domain routing.

3.5.1 Intra-Domain Routing Policies

The most common intra-domain routing approach is the shortest-path routing based on different link metrics such as minimum hop count, minimum weights, or minimum latency. Most frequently seen link state protocols such as Open Shortest Path First (OSPF)[43] and Intermediate-System to Intermediate-System (IS-IS)[49][60], or distance vector protocols such as Routing Information Protocol (RIP) [28][42] and Enhanced Interior Gateway Routing Protocol (EIGRP)[58] are based on shortest-paths. Though the above approaches are based on static link metrices, sometimes ISPs tend to use destination-based approaches or approaches implemented with dynamic link weights.

As this study mainly uses the ISP data available at Rocketfuel Repository[54], it remains important to follow the policies it has inferred in order to obtain legitimate outcomes. Rocketfuel captures intra-domain forwarding paths based on end-to-end measurements[40][62]. Thereby, it has determined link weights which characterize observed routing based on two observations: (1) least weight path(s) is/are always selected and (2) if multiple paths are chosen then they have equal weights. Nevertheless, an occurrence of a failure might contradict with the above two observations. In order to deal with such a consequence, Rocketfuel introduces an error variable[40]. Further, it improves the accuracy of inferred weights by assuming the collaborative effect of the

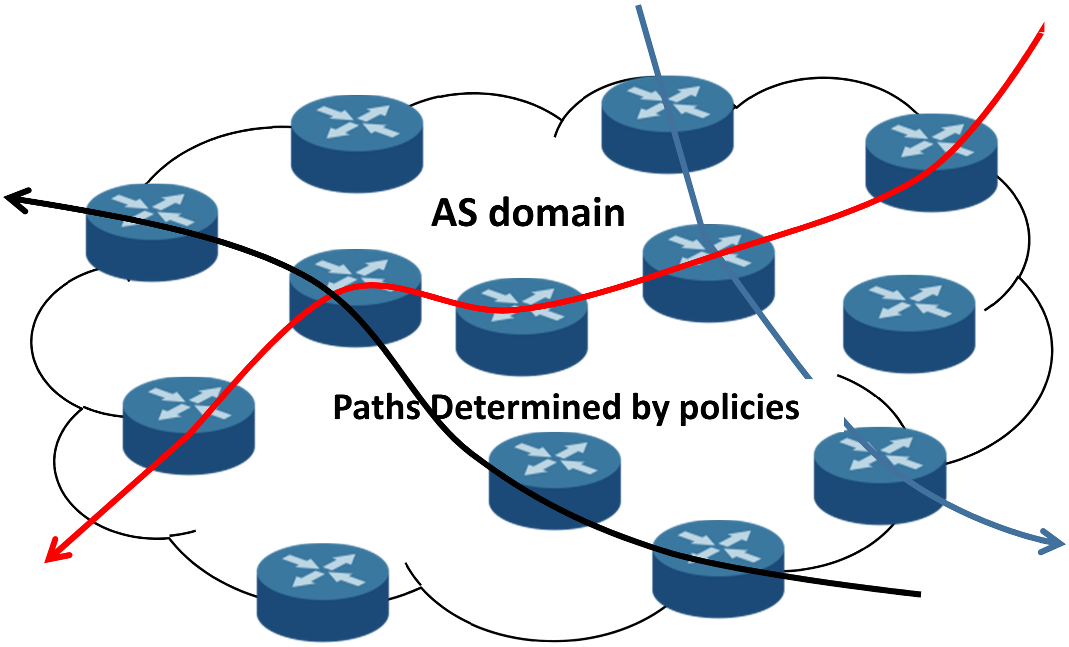


Figure 3.10: Example Intra-domain Routes Based on Routing Policies.

latency on weights[62]. Thus, a reliable set of link weights is inferred. Then the fitness of the model against the pure latency-based model and the real observed paths are evaluated[62], which proves its competence over the pure-latency model.

Considering the above factors, the simulator is designed for minimum weight based routing over the minimum latency or minimum hops based routing. Dijkstra's algorithm [20] is used in finding shortest paths in order to generate forwarding paths between different peers. Further, then peering packets to a different AS domain, the peering point is selected based on the routing policies. This situation is shown in Fig. 3.11.

3.5.2 Inter-Domain Routing Policies

Selection of the packet exit point during inter-domain routing highly influences the traffic load in an ISP domain as well as the time it takes to deliver the packet. When establishing inter-domain routing policies, peering and transiting are considered as two scenarios. This is because inter-domain routing is perceived differently depending on whether the traffic is destined to the neighbor or whether it is simply trying to take a transit to the neighbor. When the packets are destined to the neighboring ISP domain, early-exit and late-exit are two different policies commonly followed by ISPs domains. The applicability of the policy depends on the exposure of the internal topology of the neighboring ISP domain. For instance, even if there are many gateways between two ISP domains, if the source domain does not exactly know where the destination node is located in the neighboring ISP domain, it does not know the closest gateway to the

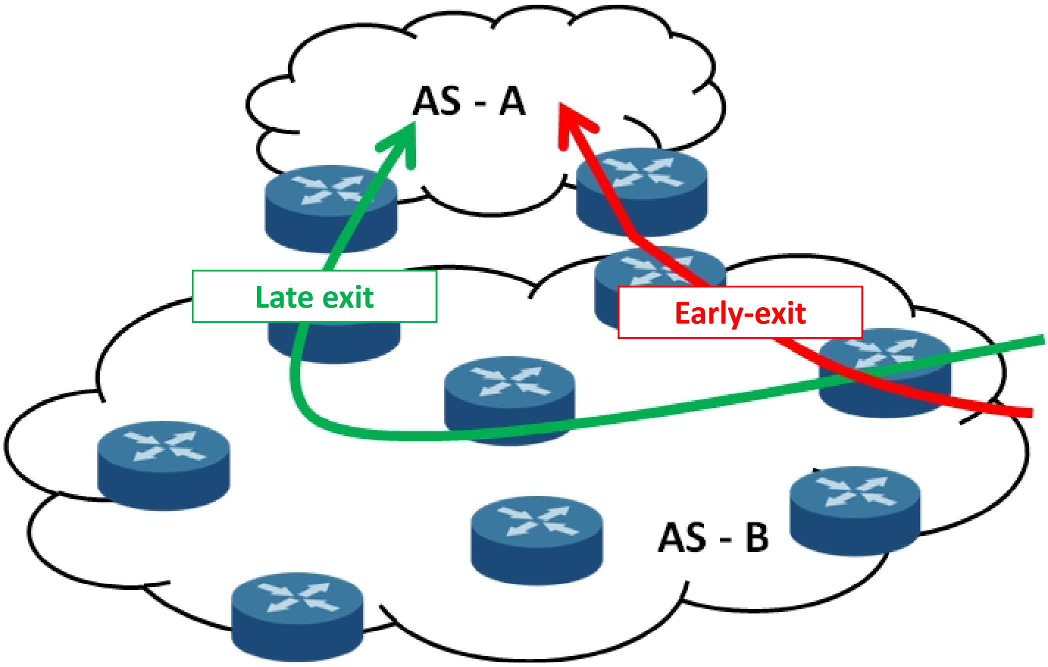


Figure 3.11: Example Intra-domain Routes Based on Peering Policies.

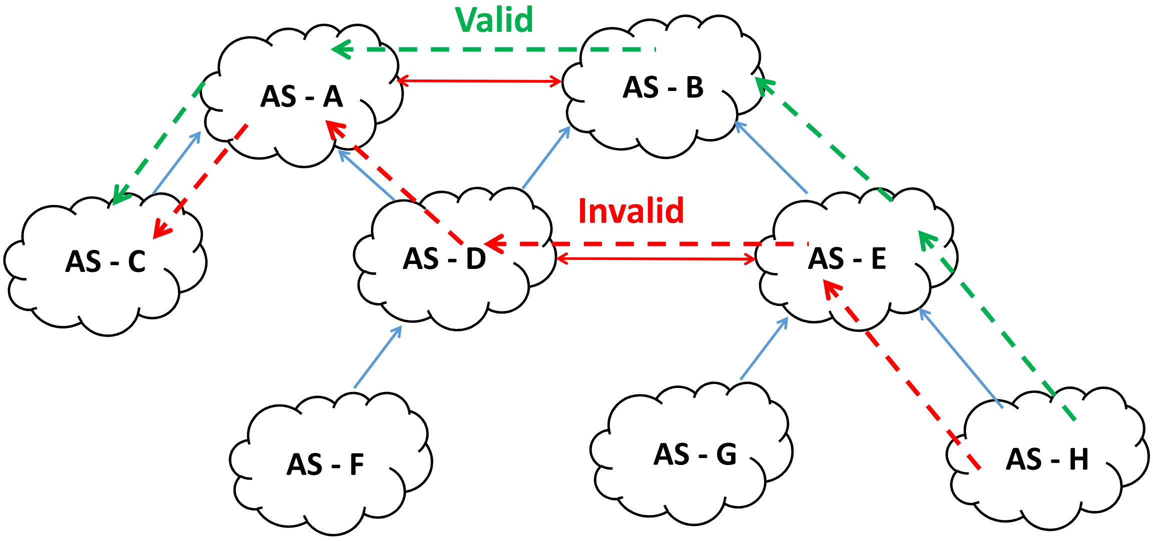


Figure 3.12: Inter-domain Routing Policies.

destination node. Usage of late-exit policy ensures routing the traffic towards the gateway that is the closest to the destination regardless of the source's location. Employing late-exit policy might increase the internal traffic load of the source ISP domain. This is because it tries to peer the data packets from the gateway which is the closest to the destination node.

In contrast, early-exit forwards traffic via the gateway closest to the source, regardless of the destination location within the neighboring ISP domain. This can cause to use a gateway residing closer to the source but farther away from the destination. This might result in a longer routing path inside the destination ISP domian.

In addition to the policies discussed above, there might be policies considering load-balancing, which allows better utilization of links. It depends on the traffic condition of the ISP domain at a given moment.

In situations where source and destination ISP domains are not neighboring, it might require to route traffic across a few different ISPs. In such situations, the ISP to which the packets should be forwarded is influenced by the relationships between ISPs. Such relationships are mainly builtup on common business agreements as already introduced in Sect. 3.4. The impact of the AS relationships in packet delivery is depicted in Fig. 3.12. According to Fig. 3.12, *AS-A* is a provider of *AS-C*. AS-E is a provider of *AS-H*. Assume that a packet originated from *AS-H* should be delivered to a destination which resides in *AS-C*. The *peer-to-peer* link between *AS-E* and *AS-D* cannot be used during this packet delivery. This is as a result of the negotiation between *AS-D* and *AS-E* about the usage of the link between them. It can be used only to exchange packets originated and destined to themselves or customers of *AS-D* and *AS-E*. However, *AS-C* is not a customer of *AS-D*. Thus, the *peer-to-peer* link between *AS-D* and *AS-E* cannot be used. In contrast, the link between *AS-B* and *AS-A* can be used during this packet exchange. This is due to the fact that ultimately, *AS-C* and *AS-H* are customers of *AS-A* and *AS-B*, respectively. However, there can be exceptions depending on whether the destination can be actually reached by using the next AS hop. Preference priorities of paths in real AS-level routing are,

1. Paths through customers
2. Paths through peers
3. Paths through providers

These aspects are considered in determining shortest paths between ASes. Thus, a modified version of Dijkstra's algorithm is used to determine the shortest AS paths between ASes. Modification is made in order to generate two types of shortest paths. One for the customers of the selected AS. The other one is for the non-customers. Thus, an AS is expected to advertise these two types of next hop information based on whether the receiver is a customer or not.

3.6 Mobility Management Entities

Location of MM Entities (MMEs) determines the efficiency of MM schemes. For instance if the MME is installed in a remote location of the topology, it may lead to excessive delays and detours. If the MME is installed in an extremely congested location, it may result in worst traffic conditions. Eventually, it can trigger packet losses. Likewise, it is important to identity appropriate locations to install MME to run MM schemes effectively.

In order to identify candidate locations for MME installment, first, it is extremely important to identify the topology and the important properties of each node (AS or router). These properties may help to determine the significance of a node and the role it plays in packet routing. For example, the following indices may include: position of the node in hierarchical topology, number of nodes it is attached to, and the amount of traffic it handles.

Two different approaches can be adopted in order to determine such properties.

- Testing the real network using significant number of test scenarios and track the behavior of all the nodes

- Mathematical analysis of test scenarios

Testing the real network will be an extremely time consuming and complicated. One reason is that it should have an enough number of test scenarios to identify the real behavior of all the nodes. The other reason is that it is practically impossible to perform such a test. Thus, the most reliable and least costly method is mathematically analyzing the behaviors of different nodes.

For the inter-AS topology of the Internet, the measurable parameters are the total number of edges of each node (degree), the number of AS hops, and the communication cost. On the other hand for the intra-AS topologies, degree, link weights, and latency can be considered as parameters. Thus, the indices that are used must be able to identify the nodes based on these measurements.

3.6.1 Mathematical Indices

3.6.1.1 Degree Centrality

According to [19] [69], in a certain network, nodes which bare much associations with other nodes might have an advantage over the others since having many connections implies that it has alternative ways of routing packets through. Further, it has much access to others directly and secondarily and so on. On the other hand, it emphasizes that it may act frequently as a third-party for other communications. Therefore, this can be thought of as a simple, but often much effective measure in terms of understanding a centrality of a given node over the others. Figure 3.13 shows the idea of degree centrality.

In the sense of hops, though this measurement seems to be more important when the cost is concerned, this measurement may fail because the connections might bare higher costs. Further, if the position of the node is skewed from the center, it will only be closer to a certain partition of the topology, which may ultimately make the node not much important though it bares multiple connections.

Degree centrality of node n_i, $C_D(n_i)$, can be calculated using the following formula

$$C_D(n_i) = d(n_i) = x_{i+} = \sum_j x_{ij} \tag{3.1}$$

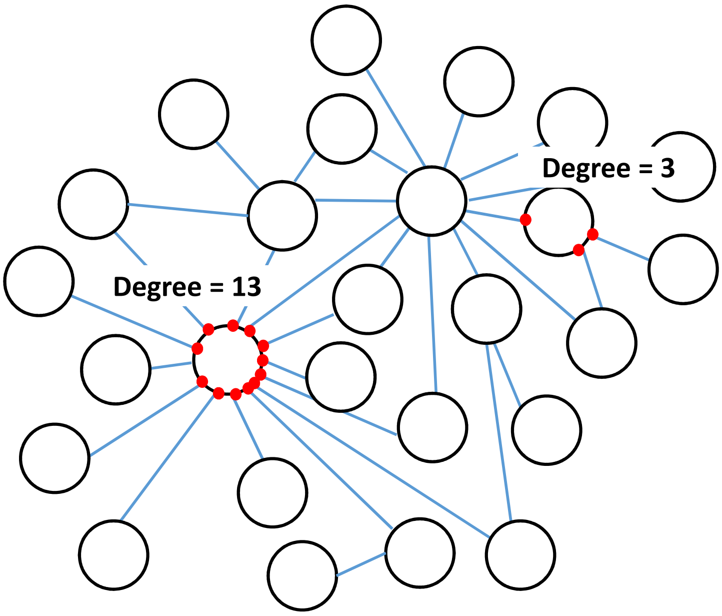


Figure 3.13: Degree Centrality.

where n_i represents the ith node. $C_D(n_i)$ stands for the degree centrality of node n_i, which is sometimes denoted as $d(n_i)$. x_{ij} is the connectivity of node n_i and node n_j. x_{ij} is 1 if there exists a link between node n_i and node n_j or 0 otherwise. x_{i+} denotes the connectivity of node n_i with the rest of the nodes in the topology. Therefore, the sum of x_{ij} over all possible j 's will result in degree centrality.

Standardized or normalized form of the above equation in order to be compared with different topological layout possibilities is as follows,

$$C'_D(n_i) = \frac{d(n_i)}{N-1} \tag{3.2}$$

where $C'_D(n_i)$ stands for the standardized degree centrality of node n_i and (N-1) is the total number of connectivities possible for a single node where N represents the number of nodes. The number, therefore, is one less than the number of nodes in the topology where it assumes the maximum possibility of connections results in connecting one node with the all the other nodes in the topology.

Formula 6.4 is equivalent to

$$C'_D(n_i) = \frac{\sum_j x_{ij}}{N-1}. \tag{3.3}$$

Since the Internet obeys the small world phenomena, in contempt of drawback of less informative, simple degree centrality can be considered as a fair measurement for understanding nodes of the topology.

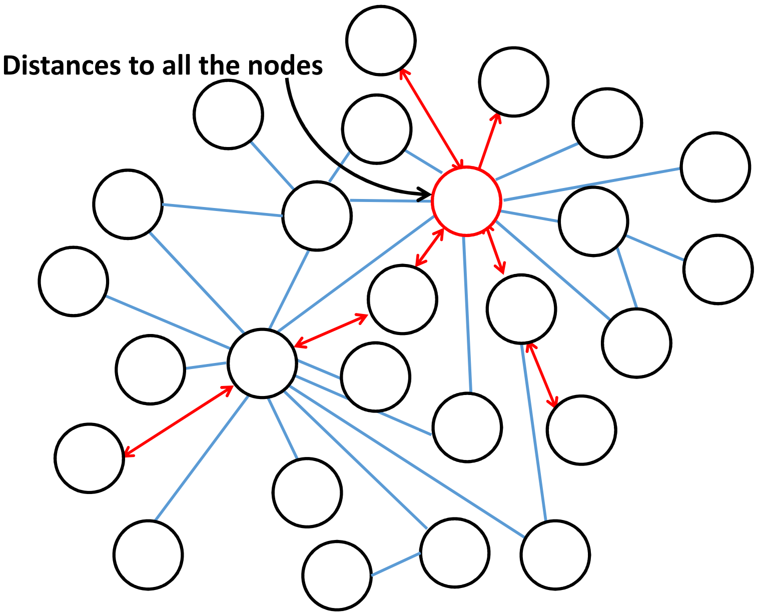


Figure 3.14: Freeman's Closeness Centrality.

3.6.1.2 Closeness

Closeness [9] [19] [27] is yet another measurement that can be used in understanding the importance of a certain node. Basically, closeness can be calculated in two major means: one is based on hops and the other is based on cost. When the closeness is considered based on hops, it describes the hop distance of the rest of the topology from the considered node. Additionally, if it is calculated considering the cost, it describes the cost distance for all the other nodes in the topology.

Closeness can be calculated using several approaches. One approach is to calculate all the hops and cost values for all the possible paths using a threshold criteria for all the end nodes in the network. This is quite costly in both time and space complexities for large topologies. Freeman's closeness[19] is based on the concept which sums the minimum outlay for each and every possible communications which is termed as all pair hop counts and all pair cost, respectively. Figure 3.14 illustrates an example scenario.

Reciprocal mean of the distribution gives the closeness value, where higher means result in lower values. Closeness based on hops, $C_{c_{(hops)}}(n_i)$, and closeness based on cost, $C_{c_{(cost)}}(n_i)$ of node n_i are calculated as follows,

$$C_{c_{(hops)}}(n_i) = c_{(hops)}(n_i) = \frac{1}{\sum_j x_{(hops)ij}/N} \tag{3.4}$$

and

$$C_{c_{(cost)}}(n_i) = c_{(cost)}(n_i) = \frac{1}{\sum_j x_{(cost)ij}/N} \tag{3.5}$$

where n_i and N represent ith node and the number of nodes in the topology, respectively. $x_{(cost)ij}$ and $x_{(hops)ij}$ represent the jth row element of the matrix along the ith column, when i is the considered element for the calculation.

In general, this value is standardized in order to be a normalized value. It follows the following formulas

$$C_{c_{(hops)}}(n_i) = c_{(hops)}(n_i) = \frac{N-1}{\sum_j x_{(hops)ij}/N} \tag{3.6}$$

and

$$C_{c_{(cost)}}(n_i) = c_{(cost)}(n_i) = \frac{N-1}{\sum_j x_{(cost)ij}/N} \tag{3.7}$$

which are simplified to be

$$C_{c_{(hops)}}(n_i) = c_{(hops)}(n_i) = \frac{N(N-1)}{\sum_j x_{(hops)ij}} \tag{3.8}$$

and

$$C_{c_{(cost)}}(n_i) = c_{(cost)}(n_i) = \frac{N(N-1)}{\sum_j x_{(cost)ij}}. \tag{3.9}$$

Formally, they show that the total outlay of a certain node is directly proportional to the closeness inversely. They can be re-calculated according to following formulas

$$C_{c_{(hops)}}(n_i) \propto \frac{1}{\sum_j x_{(hops)ij}} \tag{3.10}$$

and

$$C_{c_{(cost)}}(n_i) \propto \frac{1}{\sum_j x_{(cost)ij}}. \tag{3.11}$$

When the outlay of the calculation is considered, it is higher than that of the degree centrality calculation. But closeness calculation has the advantage that the matrix retained which holds all pair minimum hops and costs can be used in further calculations. Therefore, the opportunity cost is tolerable in the means of re-usability.

3.6.1.3 Betweenness

Betweenness [10] [45] brings up the idea that nodes that appear on many shortest paths between other nodes have higher betweenness than those that do not appear. This calculation remains very costly as it needs to obtain all the possible shortest paths between each and every node in the whole topological layout. Though there are several attempts to bring up suiting approximations, none of them are proved to be entirely accurate. This leaves this calculation much complicated and time and space consuming. An example of the betweenness centrality is depicted in Fig. 3.15

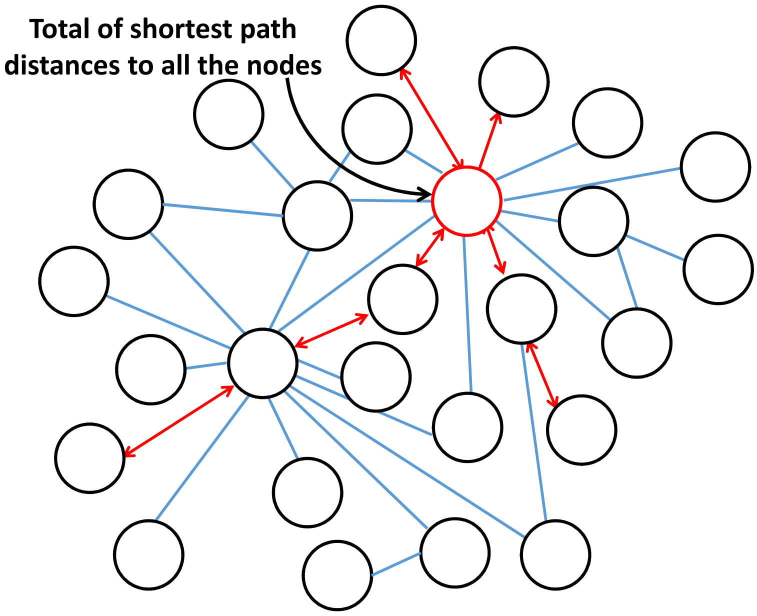


Figure 3.15: Betweenness Centrality.

Betweenness $C_B(n_i)$ of a certain node n_i can be stated as

$$C_B(n_i) = \sum_{j,k} \frac{\sigma_{jk}(n_i)}{\sigma_{jk}} \tag{3.12}$$

where σ_{jk} stands for the number of shortest paths between the node pair j and k and $\sigma_{jk}(n_i)$ is the number of paths in which node n_i appears. The betweenness is the summation over the whole topology for a certain node of this fractions.

The fact that this value is high, implies that a certain node lies in considerably many shortest paths in the topological layout and at the same time that implies the importance of a certain node when the other nodes communicate with each other.

Since this method should calculate all the possible shortest paths, when the number of nodes are very high, for example, when it is applied to the Internet, it is much time consuming. On the other hand, even when the data is reused, the extraction process of information will not be very convenient since it will have the shortest paths, not any relative cost or hops based numeral value. Therefore, the opportunity cost of using this method remains questionable. Further, for topologies which follow the small world phenomena, there is a high probability that those nodes which do not appear in the shortest path to have a closer distance in the means of hops and cost both. That occurs when that node might be directly connected to a node which appears in the path connecting the considered pair of nodes with a considerably low cost or hops. It results in having closer nodes in both cost and hops aspects, which do not appear in the shortest path but still bares a very close communication path.

3.6.2 Mobility Management Entity Placement in Inter-AS Level Mobility Scenario

In general, all the ASes are assumed to maintain their own MME. However, most appropriate secondary MME placement is expected to be determined during the evaluation detailed in Sect. 4.2. The indices mentioned above are broadly used in order to idntify prospective installations.

3.6.3 Mobility Management Entity Placement in Intra-AS Level Mobility Scenario

Installation of MMEs plays a severe role in determining performances of employed MM schemes. MME placement in Intra-AS domain should take into consideration the role played by each node within the local domain. Then, it is required to determine the fitness of reach node to act as the MME. For instance, a node resides averagely closer to all the ARs can be considered as a candidate MME. It is because all the ARs in the network lie close to that node. Thus, the communication between the MME and the AR will be least costly in terms.

Two mechanisms to determine MME installments are adopted in this study.

1. Degree and closeness-based MME installation

2. Location-based MME installation

3.6.3.1 Degree and Closeness-Based Mobility Management Entity Installation

Degree centrality and the closeness centrality are used to determine the prospective MME locations. First, routers with the upper 10% of degree centrality is selected. Then, the selected set is sorted considering the closeness centrality. Finally, the required number of MMEs are selected based on the rank. This MME installation determines the prospective locations solely based on the router's accessibility considering the whole local domain.

3.6.3.2 Location-Based Mobility Management Entity Installation

The location-based MME installation tries to address the real world scenario of MME selection. Currently, the network architecture follows a hierarchical structures. General ISPs tend to build hierarchies following geography. Thus, it is identified that it more realistic to determine MMEs based on the geographical locations. Therefore, the location-based MME installation sorts the routers based on the geolocations. Then, it determines prospective MME installation based on degree and closeness centralities for different geolocation blocks.

3.7 Mobility Patterns of Mobile Nodes

In order to determine the fitness of the DMM schemes, it is required to determine realistic test scenarios. For instance, it is extremely unlikely for a MN to access the Internet from geographical points which lie far apart during its subsequent attachments.

3.7.1 Mobility Patters in Inter-AS Level Mobility Scenario

Inter-AS mobility patterns are determined based on the ideology that every AS hosts MNs. However, for mapped topologies, when the AS types information are available, such information is used to determine whether that AS is a candidate location for the MN to access the Internet. For instance, if the AS is a solely transit provider, then, it is extremely unlikely that it hosts MNs. The types of ASes are as follows,

- Transit providers : Only provides transit services for other ISPs / ASes (mainly tier-2 ISPs).
- Access providers : Leaf ISPs/ ASes that host MNs.
- Transit/Access providers : Provides transit services for other ISPs and hosts MNs.
- Content Hosts : Hosts content. Mainly service and storage oriented (E-mail, web-hosting, and online storage).
- Enterprise : Dedicated for enterprises.
- Educational / Research : Reserved for educational networks and reserach purposes.
- Non-profit : Hosted by nonprofit organizations.

This classification is based on the information found at CAIDA [12].

When the initial location is determined, then the arbitrary relocation ASes are determined mainly based on the AS relationships and the distance between ASes. Two such scenarios are depicted in Figs. 3.16 and 3.17. In these two scenarios, the relationship between *A* and *B* is different. When movement within 2 AS hop distance is considered, the possibility to select an AS as a candidate handover location is determined considering the same criteria defined for the routing. *Customer-to-provider* are likely to provided paid accesses services. Therefore, movement is considered to be probable among customers of a common AS (Fig. 3.16- common provider *B*). *Peer-to-peer* relationships are considered as least supportive in providing collaborative access to MN. Thus, it is considered least probable (Fig. 3.17).

3.7.2 Mobility Patters in Intra-AS Level Mobility Scenario

In order to generate mobility patterns, first a random set of access points are initially selected as the default residential locations of the MN, based on the number of ARs in the selected topology. Then the movements of those MN nodes are determined based on two criteria.

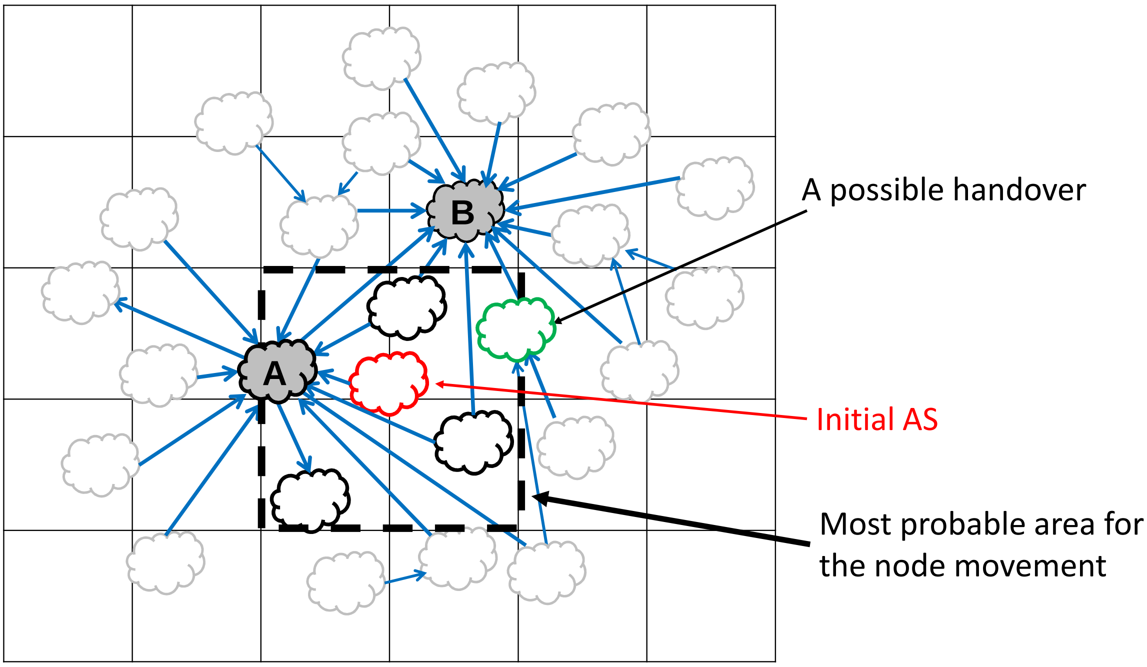


Figure 3.16: Mobility Pattern Determination - Possible Handover.

- Random movement

- Selective movement

In case of random movement, the subsequent locations to move are selected completely randomly. In contrast, the selective movement scenario takes into cosideration the real world movement of the users of MNs. Mobile nodes in the Internet can be considered to be synchronized with the movement of users. For example, it is extremely unlikely to expect a user to suddenly move across a continent and connect to the Internet promptly. Thus, the selective moment scenario draws realistic mobility patterns for the MNs for a sustainable analysis.

For instance, the Telstra (Australia) ISP network has a considerable amount of ARs allowing a larger set to be selected whereas Tiscali (Europe) has the least. Then considering the geolocations of those initial locations, some feasible mobility patterns are generated, i.e., up to four handovers are assumed for each scenario defined by each initial location, which covers the initial location itself and four more candidate locations closer by which are determined based on proximity.

A unit called the *proximity factor* and a unit distance called *y* are defined to determine the area which is considered in deriving the set of handovers. The *proximity factor* can take values 1 through 4 and the value *y* is determined based on the area covered by the ISP, which is around 1/8 of the minimum of height or width of the area covered by the ISP. For example the Telstra (Australia) ISP network covers Australian mainland

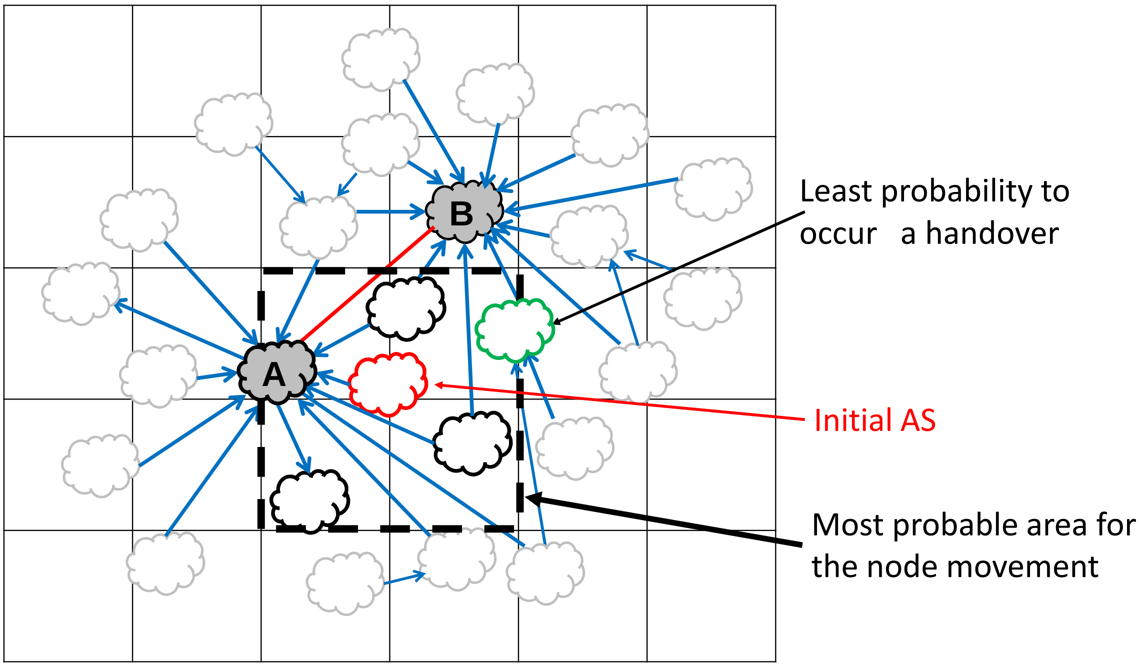
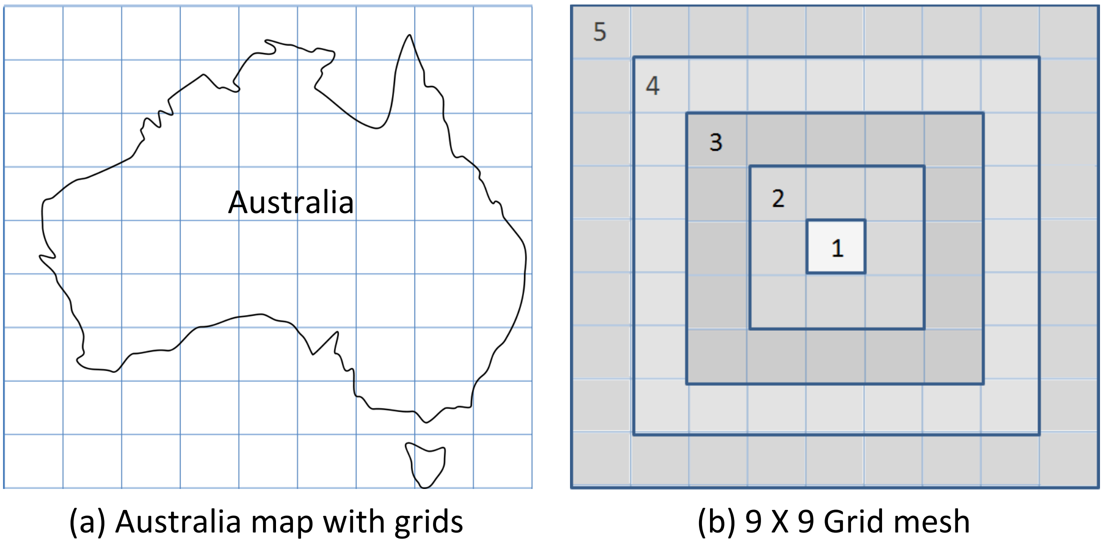

B
A
Least probability to occur a handover
Initial AS
Most probable area for the node movement

Figure 3.17: Mobility Pattern Determination - Least Probable Handover Scenario.



(a) Australia map with grids

(b) 9 X 9 Grid mesh

Figure 3.18: Map of Australia with Grids and the 9X9 Grid mesh.

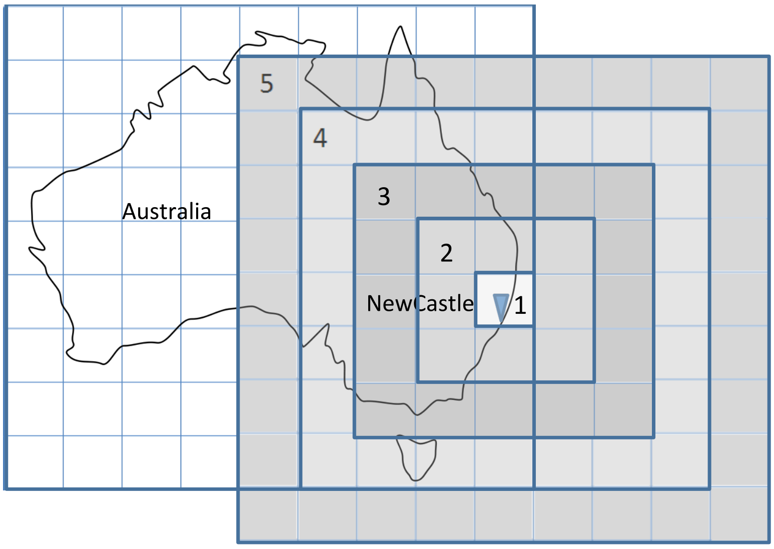


Figure 3.19: Mobility patten determination using the Grid mesh.

which is approximately 4,100 km × 3,200 km, if Tasmania is excluded. Thus the distance factor y is defined to be around 400 km which is closely 1/8 of the height. Then the map of Australian mainland and Tasmania is being meshed with a square grid, each square covering an area of 400 km × 400 km. The resultant map has 9 × 9 grids as shown in Fig.3.18-(a) because the Tasmania island is included. A separate 9 × 9 grid mesh, which is depicted in Fig.3.18-(b), is used in determining consecutive locations of the MN.

Initially, when a certain location is selected as the initial location to place the MN, the 9 × 9 grid mesh (Fig.3.18-(b)) is placed over the map (Fig.3.18-(a)) such that the central square of the 9 × 9 grid mesh overlaps with the square which contains the selected location within the map. This is elaborated in Fig.3.19. Each time the *proximity factor* is increased, the distance y is multiplied by the proximity factor to widen the area covered. i.e., when the factor is 1, only the central square labeled with *1* in Fig.3.19 is considered. Next when the factor is 2, the eight connected neighborhoods of the central square are included to the area of consideration which is marked by *2* in Fig.3.19.

Similarly, consequent squares are attached to the considered area as the proximity factor increases. For example, assume that a certain AR which is randomly picked from the Telstra (Australia) ISP network resides in Newcastle, New South Wales as shown in Fig.3.19. Then, primitive movements are assumed to be mainly scattered in New South Wales, specially in the eastern area. Or in case that the selected MN initiates in South Port, Queensland, then the probability of movements will be intense around south-east Queensland and north-east New South Wales. Thus, in cases where there is a limited number of ARs within the defined proximity factor, as in Sprint (USA), the proximity factor is gradually increased to find a feasible data set. However, the increment of proximity factor is bounded to a certain extend to make the selection realistic and thus if the boundary is reached before selecting four successive locations to relocate the MN, the number of handovers will be restricted as a result. This is observed in Tiscali (USA). The range of proximity factors that are used in defining the areas within each ISP are shown in Table 3.3.

3.8 Sessions

The session defines the CN with which the MN communicates using its IPv6 address which is enabled to benefit from mobility. However, there can be two types of sessions in general. Ones that do not require mobility at the new location and ones that require MM. These two groups are defined featuring the concept of *packet offloading* which is adopted by some ISPs. Providing MM unnecessary will introduce unnecessary control and data-plane overhead. Thus, the *packet offloading* procedure lets the ISP to offlaod the traffic which do not require mobility. Thus, it improves the performance of the network. However, it requires to identify the sessions that can survive an IPv6 address change. Some of the MM schemes facilitates *packet offloading*. Thus, it is also considered when designing the simulator.

In inter-AS and intra-AS evaluations, both sessions where the MN corresponds with local CNs and sessions during which the MN corresponds with external CNs are considered. When the CN resides in the same ISP as the MN, intra-domain policies determine the route during the intra-domain routing. In case where the CN resides in an external ISP, the AS hops taken are determined based on the inter-as routing policies and the relevant PPs are selected based on the inter-domain policies.

3.9 Traffic Pattern

A simple traffic pattern is assumed in constructing the simulator. Sessions are established at each location according to the generated session data set and data packets are sent at a constant rate triggered by the initial connection. After the detachment, the CN is presumed to send packets at the same rate and thus the number of packets sent during the handover will be proportional to the handover time. After the successful completion of the handover, delay and cost in receiving the first packet would therefore be consistent with the time it takes to deliver the packets buffered by the MME if there are any.

However, the simulator is extendable to allow more complex traffic patterns. As the intend evaluations do not require complicated traffic patterns at this level, it is considered as out of scope.

3.10 Performance Measurements

The basic measurements used to evaluate the DMM schemes are as follows:

- Routing Cost
- Routing Latency (only for intra-AS scenarios)
- Total number of hops (AS hops for inter-AS scenarios or router hops for intra-AS scenarios)
- Node Load (AS load for inter-AS scenarios or router load for intra-as scenario)

All the selected DMM schemes are evaluated over one topology at a time using synthesized movement patterns, where the MN is assumed to bare a random set of sessions that were drawn according to the descriptions given above in Sect. 3.8.

For any communication, the *Routing Cost* is calculated based on the AS communication cost for inter-AS scenarios. For both synthetic and mapped topologies, these measurements are available.

For intra-AS scenarios, it takes into consideration the link weights determined by BRITE and Rocketfuel for the synthetic and mapped topologies, respectively. These weights reflect the effects of topological and policy constraints in mapped intra-AS topologies. Costs for control and data planes are the sum of link weights traversed by controlling signals during handover and the sum of weights of links traversed by the first data packet, respectively.

The second measurement of the *Routing latency* is also calculated following the same criteria as in cost. However, the AS-level topology of the Internet does not have a latency based information for inter-AS communication. Thus, this measurement is not feasible for the synthetic and mapped AS-level topologies.

After determining the paths that should be taken in each test scenario, the number of hops that should be traveled in order to complete the control/data-plane procedure is counted and reported as the *total number of hops*. It gives another perspective to the routing cost, since even though the cost is lower, a packet might have to take a number of hops in order to arrive at the destination. This can be considered as another measurement to detect detours that have been made. All the synthetic and the mapped topologies contributes to this measurement positively.

During the simulator construction, the *Node Load* is assumed to be proportional to the number of packets forwarded by the AS or the router. AS load implies the number of control / data-plane packets that enters a certain AS. It can be considered as one of the measurements to give a good idea about the traffic that occur in a certain AS. On the other hand, the router load specifically identifies the traffic handled by each of the router. This functionally is also extended to identify the event loads handled by each MMEs,

3.11 Implementation and the Functionality of SimNet-DMM

Using the concepts described earlier in this chapter, a specific simulator is written in C language which is capable of simulating the packet flow over IP when different mobility protocols are employed. Since the aim of this simulator is to compare performances of different DMM schemes, the rest of the protocols in the protocol stack is assumed to be static. The basic functional flow of the simulator is given in Fig.3.20.

First, the main input files, topology and policy data, are fed to *Topology Simulator* (Fig.3.20 (1)). It populates two 2D matrices with unprocessed latencies and costs / weights of links. As already mentioned, the inputs varies depending on the topology used. Then, *Shortest Path Generator* is triggered (Fig.3.20 (2)). The shortest path generation is based on the Dijkstra algorithm[20]. A slight modification is introduced

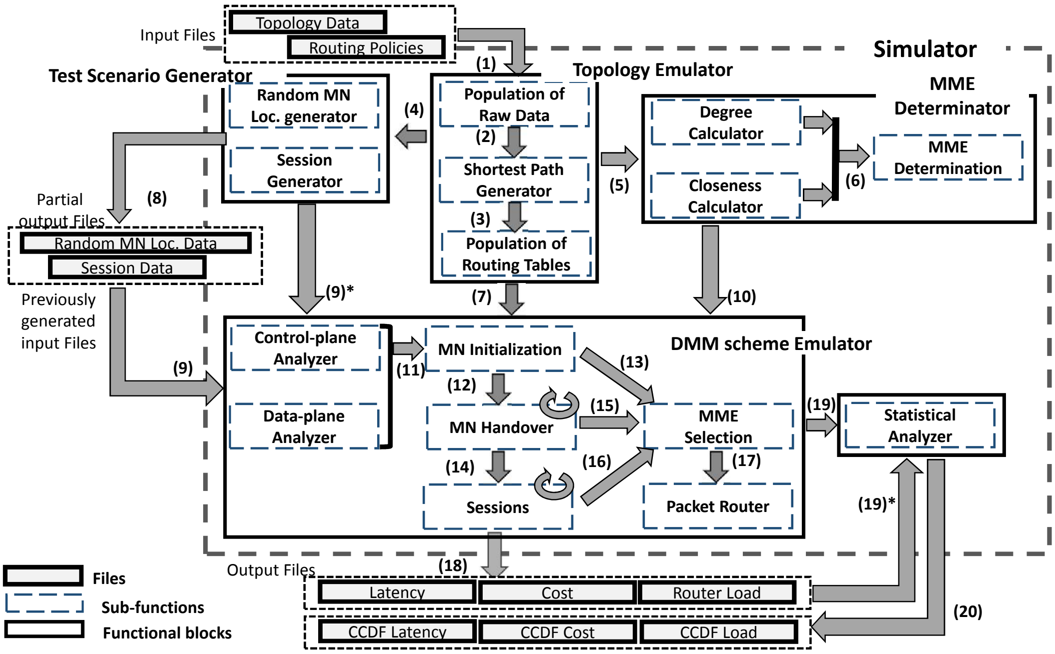


Figure 3.20: Architecture of the Simulator.

to the Dijkstra algorithm in order to accompany routing policies for mapped AS-level and ISP topologies.

It triggers *Population of Routing Tables* (Fig.3.20 (3)), which populates two separate 2D matrices with total latency and cost of shortest paths between all the routers in the network. The 2D matrices are calculated in an incremental manner. This completes the routing table population.

Next, the data feed (Fig.3.20 (4)) triggers *Random MN Location Generator* and *Session Generator* in the functional block *Test Scenario Generator*. It randomly selects required number of test sets fed at the run time of the simulator. If the roles of the ASes or routers are available, this function makes use of those roles to determine the MN locations. The selection includes two ARs for the MN‘s initial attachment and re-attachment as described as determined by the evaluation scenario. The simulator is capable of generating partial output files with generated intermediate data (Fig.3.20 (8)). This optional functional box may be disable in order to allow the previously generated files to be re-used. Those pre-generated files can be fed to the simulator directly (Fig.3.20 (9) instead of Fig.3.20 (9)*).

On the other hand, the raw data (Fig.3.20 (5)) triggers *MME Determinator*. It calculates *Degree Centrality* and *Closeness Centrality* using respective functions. *Degree Calculator* calculates the number of links per each router, whereas *Closeness Calculator* computes the latency-based closeness centrality for all the routers in the topology using the intermediate data found in the routing tables. Closeness is determined hops or cost/weight.

Completion of the above two functions triggers *MME Determination* function (Fig.3.20 (6)). It is responsible for identifying MME installment locations. At this point, only a tentative list of candidates are generated. Thus, it is required to determine all the candidate locations to install MMEs for different DMM schemes prior to emulating the DMM schemes.

After the successful completion of above steps, the simulator triggers the main functional block *DMM Scheme Simulator* (Fig.3.20(7)). It is triggered by feeding *Random MN Location Data*, *Session Data* (Fig.3.20 (9)), and *MME Data* (Fig.3.20 (10)). Thus, all the above functions play a major role in the efficiency of the *DMM Scheme Simulator*. *DMM Scheme Simulator* consists of selective sub-simulators. Each sub-simulator implements a distinctive DMM scheme. This is because the similarities between the DMM schemes that are found in the literature are extremely low. These sub-simulators are executed one at a time which includes *Control* and *Data-plane Analyzers* which is specific for the selected DMM scheme. These analyzers module the flows defined by the specification of the selected DMM scheme.

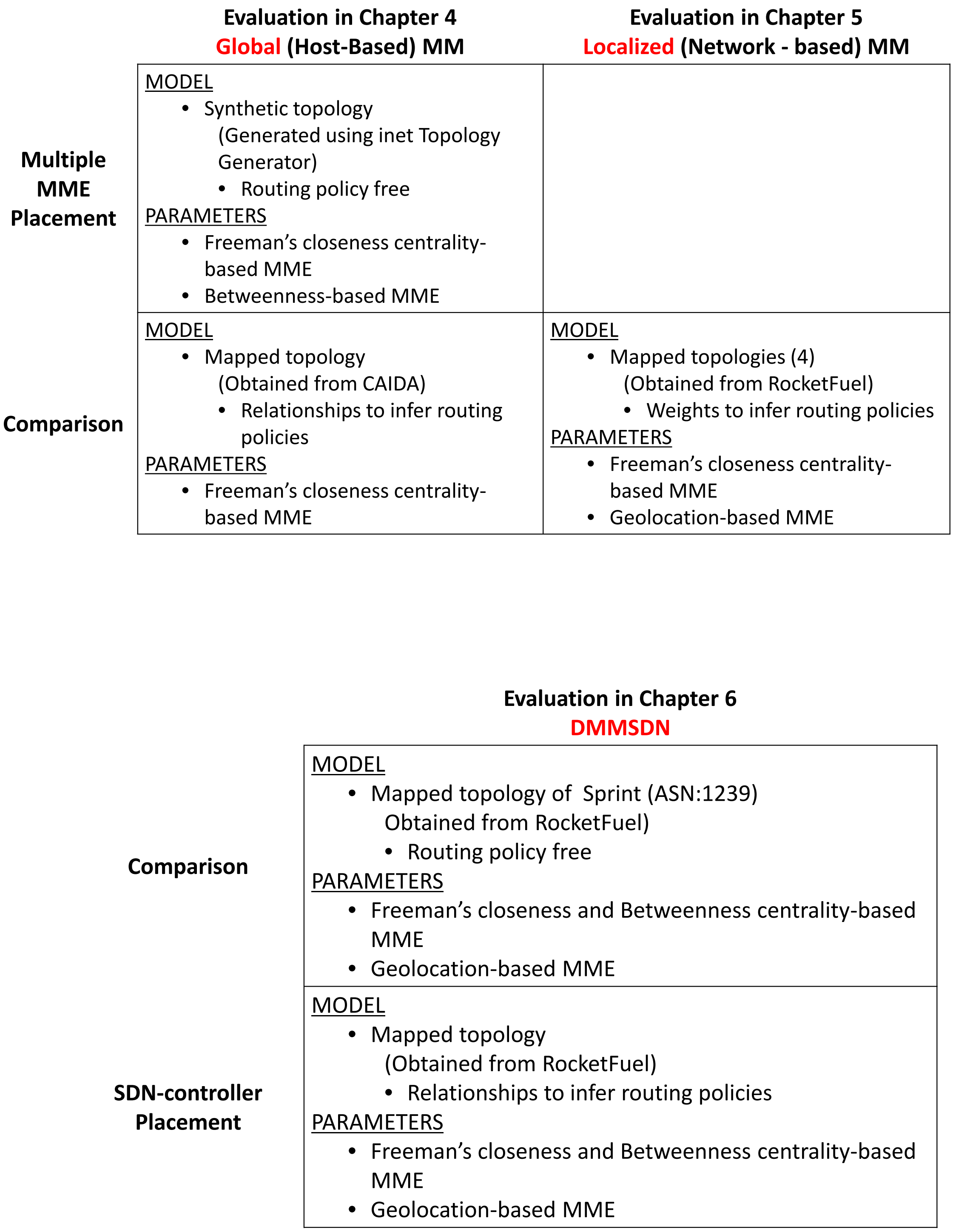
In order to allow the future DMM schemes to be simulated, the module to simulate the DMM scheme is kept as simple as possible. Thus, the only alteration that should be made to accompany a different MM scheme is to inherit the general control and data-plane analyzer and modify it to reflect the required MM scheme.

Control and data-plane analyzers defined above test each test scenario, starting with *MN Initialization* (Fig.3.20 (11)). Considering each successful handover with *MN Handover* (Fig.3.20 (12)) the session continuity of the *Sessions* (Fig.3.20 (14)) are tested. During each scenario, if required, *MME Selection* (Fig.3.20 (13),(15),(16)) is taken place. This selection is based on the requirements specified by each MM scheme. The tentative list which was already created which has the candidate locations for the MMEs will be used and the selection will be done accordingly. Finally, *Packet Router* imitates the packet flows in both control and data-planes (Fig.3.20 (17)).

As a consequence, the resultant output files, *Total latency*, *Total cost*, and *Node load*, are generated (Fig.3.20 (18)). The data feed (either Fig.3.20 (19) or Fig.3.20 (9)*) triggers an optional function *Statistical Analyzer*. It summarizes the above results and generates respective files containing Complementary Cumulative Distribution Function (CCDF) data (Fig.3.20 (20)). Similar to *DMM Scheme Simulator*, this function also accepts external input files (Fig.3.20 (19)), which are previously generated

3.12 Models and Parameters Used in this Study

There is a considerable amount of models and parameters discussed in this chapter. However, all of them are not used in the evaluation at once. Figure 3.21 summarizes the usage of main models and parameters in the subsequent chapters. As only the main parameters are outlined in this section, Chapts. 4, 5, and 6 should be referred for extended descriptions.



	Evaluation in Chapter 4 Global (Host-Based) MM	Evaluation in Chapter 5 Localized (Network - based) MM
Multiple MME Placement	MODEL • Synthetic topology (Generated using inet Topology Generator) • Routing policy free PARAMETERS • Freeman's closeness centrality-based MME • Betweenness-based MME	
Comparison	MODEL • Mapped topology (Obtained from CAIDA) • Relationships to infer routing policies PARAMETERS • Freeman's closeness centrality-based MME	MODEL • Mapped topologies (4) (Obtained from RocketFuel) • Weights to infer routing policies PARAMETERS • Freeman's closeness centrality-based MME • Geolocation-based MME

	Evaluation in Chapter 6 DMMSDN
Comparison	MODEL • Mapped topology of Sprint (ASN:1239) Obtained from RocketFuel) • Routing policy free PARAMETERS • Freeman's closeness and Betweenness centrality-based MME • Geolocation-based MME
SDN-controller Placement	MODEL • Mapped topology (Obtained from RocketFuel) • Relationships to infer routing policies PARAMETERS • Freeman's closeness and Betweenness centrality-based MME • Geolocation-based MME

Figure 3.21: Models and Parameters Used in the Study.

3.13 Limitations of SimNetDMM

In general, the simulator, SimNetDMM makes some basic assumptions in simulating the mobility scenarios. It assumes that the processing and the residence times of all the routers are negligible when compared to the routing time. However, this scenario is not legitimate for the AS nodes, since after a packet enters an AS, it might lake a long time to exit the AS. Unavailability of latency information for ASes relieves this issue. Thus, this assumption is not made for the AS-level topology.

When determining the packet flow, it is assumed that User Datagram Protocol (UDP) based applications are running on the application layer. That is no control overhead due to transport layer is introduced. Further, *DMM Scheme Simulator* does not support variable packet sizes, thus constant packet sizes are determined for control and data packets beforehand, which can accommodate all the selected MM schemes. Further, the focus of this study is to determine the performance of the MM schemes exclusively. Thus, the background traffic and the bandwidth limitations are not taken into account when constructing the simulator. However, in mapped ISP topologies, the weights derived by the Rocketfuel [41] [63] [64] contain the constraints of path determination. These weights reflect routing policies determined by the administrator. This includes bandwidth restrictions and other administrative decisions that result in choosing the observed path. Thus, the usage of the weights given in Rocketfuel to determine paths, is expected to capture the route inflations due to background traffic and bandwidth limitations.

3.14 Reliability of SimNetDMM

SimNetDMM is an L3 simulator which does not assume any IPv6 route influences or inflation due to the rest of the network protocol stack. Thus, the realiability of the results will inflate due to extreme affects that might avert the underlying routing. For instance, this model assumes that all the links are up and available to be used. The only deterministic factor of routes is considered as the source and destination IPv6 addresses, and the routing policy applied when the communication takes place between those two entities. This routing policy is derived based on the routing observed by the traceroute servers employed by Rocketfuel in case of router-level mapped ISP topologies. In case of AS-level topology, it is the relationships defined between ASes. Thus, the reliability of modeled L3 is determined based on,

- Reliability of derived control/data sequence
- Reliability of the source and destination IPv6 addresses that are used
- Reliability of the paths connecting the source and the destination IPv6 addresses

3.14.1 Reliability of the derived control/data sequence

SimNetDMM tries to determine the control and data-planes of the candidate DMM schemes as a sequence of packet exchanges. First, for standard protocols, the signal and control sequences that are available in technical papers are formulated. For non-standard protocols, technical papers describing the signal and data flows are used to formulate the behavior of the scheme. Most of the selected schemes are published as related Internet drafts. Thus, they have sketches with tentative control/data-plane behavior and the functionality. However, if there is no defined approach for the control and data-planes other than the idea, then the control and data sequences are constructed using minimal overhead. It is done by considering already available protocols. Thus, the reliability of the defined control and the data-planes is extremely dependent on the assumptions and suggestions that are made by proposals. It is due to the fact that the main aim of this simulation is not to confirm the correctness or the functionality of the DMM scheme. Instead, the focus is to test the applicability of the proposed idea in terms of expected control/data-plane performance.

3.14.2 Reliability of the used source and destination IPv6 addresses

SimNetDMM modeled control and data-plane communications solely consider the IPv6 addresses of the source and the destination. The source and the destination that are determined for each control or data communication that takes place in the network are used as evaluation parameters. The selection of these parameters are conducted considering the real properties of the nodes. For example, the ARs are selected from known set of ARs, which are observed at the mapped phase. Thus, the reliability can be considered extremely high and solely dependent on the reliability of the data sets that are used. Further, the derived mobility patterns are considering real user movement patterns. For instance, the speed at which a device can move is extremely dependent on the real world user movement scenario. SimNetDMM considers a block of 500^2 km^2 as the most probable relocation area. Thus, it can be considered to cover the worst case realistic movements as well. Therefore, a huge deviation from the obtained results cannot be expected.

Further, the nodes to employ MMEs are also determined based on administrative perspectives. The degree of connectivity, closeness to the rest of the topology, or the appearance of the node in shortest paths like factors were considered to determine the fitness of the node to act as the MME. Thus, the placement can be considered extremely closer to the reality. Therefore, the reliability of location of the MMe can be considered extremely high.

3.14.3 Reliability of the paths connecting the destination and the source IPv6 addresses

Even though the determination of the control and data-plane behavior and the selection of each source and the destination pairs for each of those control and data-plane event are justified above, the major factor that determines the correctness of the obtained results is the correctness of the paths derived. The routes that are taken between two nodes during the simulation is an imitation of the real routes that are observed by the traceroute servers employed by the Rocketfuel for the ISP topologies. The observation of traceroute servers captures all the routing policies that are employed. Thus, the observation is the outcome after applying all the routing modifications by the network administrator. Therefore, the observed routes are expected to capture all the policies that are exerted. However, Rocketfuel confirms the reliability of the routes by comparing the weight based shortest paths with the real observations [41]. During Rocketfuel weight inference, fraction of observed paths that were least costly in terms of inferred weights is 87-100. Therefore, the path inflation is minimal when compared with the reality. The reliability of the paths generated by SimNetDMM is equivalent to the reliability of the path confirmation rate of RocketFuel. This can be considered as the conformity ratio for the SimNetDMM produced results as well.

3.15 Related Work

DMM can be considered as one of the fields which least benefits from simulator resources found in the literature. There are no specialized simulator for DMM. However, some of the generic simulators support standard MM protocols.

Network Simulator version 3 (NS-3) [46] is one of the leading open source network simulators that is found in the literature, which supports simulat ion of MM. It is widely used by the network research community. Direct Code Execution (DCE) module of NS-3 is commonly used to execute already standardized network protocols or applications. This mode does not allow source code modifications. Usagi-Patched Mobile IPv6 stack (UMIP) [47] implements MIPv6 and Network Mobility (NEMO) protocols. However, none of the other mobility protocols are supported as on 2015.

NetSim [44] can be considered as another generic simulator found in the literature. Unlike NS-3, NetSim is a commercial simulator. Being significantly old, NetSim only supports IPv4. However, most of the novel DMM schemes assumes an IPv6 environment, which emphasizes the inappropriateness. Despite of lack of support for IPv6 implementation, NetSim also has the disadvantage of not supporting alterations to the assumed mobility scheme.

OPNET [48] is also another generic simulator used by the research community. It is also a commercial simulator like NetSim. Thus, the models cannot be flexibly modified. Mobility protocols assumed are fixed where modifications are not permitted. Therefore, OPNET cannot be considered as a prospective candidate simulator for DMM schemes.

3.16 Summary

Performance evaluation of DMM schemes is merely seen in literature. The reason can be identified as the least support from generic simulators, such as Network Simulator 3 (NS-3). Generic simulators provide least support to analyze non-standard protocols. Further, they do not complement novel DMM concepts such as control/data-plane separation and distribution of control-plane. As a result of most DMM schemes being not implemented or prototyped, evaluation and comparison of such schemes remain extremely challenging.

This chapter introduced SimNetDMM, a network layer simulator that supports evaluation of DMM schemes. SimNetDMM is a performance evaluation tool, which cannot be used to confirm the functionality of the scheme. Rather, it just analyzes control and data-plane behavior suggested by a DMM scheme. As special features, SimNetDMM allows the evaluation of DMM schemes over real topologies. It considers real mobility environments such as the Internet or Internet Service Provider (ISP) networks. Then, administrative and topological constraints are taken into consideration to place Mobility Management Entities (MMEs). Further, realistic mobility patterns for Mobile Nodes (MNs) are considered. This allows the comparison of different DMM schemes using closely imitated real world scenarios.

Chapter 4

Evaluation of Global Distributed Mobility Management Schemes

As already introduced in the Chapter 2, host-based MM schemes require mobility management to be triggered by the MN. Thus, the mobility of the MN should not be restricted to a certain network which is capable of identifying its movement. Instead, the MN recognizes its movement and it is capable of notifying its movement to the agent(s) handling mobility. Based on this consideration, host-based mobility management can be considered suitable to handle mobility of a node in a wider range, such as inter-AS node mobility.

4.1 Motivation

Providing DMM in the scope of AS-level needs negotiation between different ISPs. Thus, it is extremely important to decide the optimal places to install secondary MME. Optimally placing the subsequent ISPs will allow to increase performance of DMM scheme. Further, installing unnecessary MMEs will only introduce excessive costs. These costs will include installment cost as well as controlling overhead.

Next, there are different DMM proposals in the literature. However, the best approach is not yet distinguished. Thus, it is required to determine the best approach to handle inter-AS mobility.

4.2 Performance Evaluation of Multiple HA Placement in Mobile IP Environment

This section gives details of the evaluation that is carried out to determine multiple HA placement from the operational view point. Thus, it tries to recognize the best secondary AS to install an additional HA in the MIPv6 environment in order to improve performance. If the MN moves away from the AS keeping the session continuity, having a single HA in the home network will introduce a packet detour overhead. In the practical situation, a single AS maintains its own HA in order to facilitate the MNs of the

AS. Thus, the attempt is to duplicate and distribute the HA such that the control and data-planes obtain enhanced performance.

The first attempt is to identify the most reasonable parameter to classify the ASes. This is because the AS hierarchy described in Sect. 3.4.2 appeals that an AS will perform differently with MM scheme depending on its tier. For example, ASes belonging to *tier-1* will play a major role in routing packets between different MNs belonging to different ASes. Thus, having the HA in that AS might result in least non-optimal routes. In contrast, having HA installed in a leaf AS belonging to *tier-3* might result in a critical packet detour, if the MN moves further away from that AS.

Rather than analyzing the best secondary HA placement for each AS node in the Internet, it is contributive to identify differences between different AS nodes envisioning to categorize them. After that, a generic conclusion can be drawn for each category. Therefore, first, the location specific properties of AS nodes are identified.

A set of mathematical indices are used to obtain an idea about the ASes in the network. These indices includes simple descriptive statistics to indices like degree centrality, reachability, closeness, and Freeman's closeness. All these indices are capable of being obtained by SimNetDMM described in Chapt. 3. Finally, the impact of single and multiple HA placements are analyzed using different combinations of HAs of different groups identified.

4.2.1 Approach

Global HAHA (Sect. 2.4) and Migrating HA (Sect. 2.5) are considered as the candidate DMM schemes for the evaluation. They result in the similar data-plane routing, where duplication and distribution of HAs can be seen. By pushing these HAs closer to the MN, these schemes theoretically can reduce the non-optimal routing. MIPv6 (Sect.2.3) and MIPv6 with indirect routing (Sect. 2.3.1) are used as referential schemes. Thus, it is speculated that the optimized routing mode is not employed in the MIPv6 environment (Sect. 2.3.2) since the signalling of return routability is much complicated. This results in obvious mediation of the HA in all datagram exchanges taken place between the CN and the MN. This scenario is considered as the centralized version of the selected DMM schemes.

The evaluation is focused on global mobility. Thus, synthetic AS-level topology of the Internet is generated using the inet topology generator (Sect. 3.4.1.1). The size of the synthetic AS-level topology is determined to be 30,000 nodes (ASs). Today's internet is known to have more than 30,000 ASes.

The impact of HA placement can be majorly identified by focusing on the cost accompanied with general data packet exchange. According to the specification given in the Chapt. 3, the control and data-planes are modeled as a DMM scheme sub-emulators.

As the first step, this evaluation only concentrates on the communication cost. This is due to the fact that the signalling cost is comparatively negligible when compared with the communication cost. Thus, the aim is to identify the HA installment which optimizes the data-plane. However, the location of the MM agent will indirectly affect the signalling cost as well, as cost effective positioning of MN implies that signalling between these agents will also be minimal.

The cost of communication can be derived based on three different factors.

Table 4.1: Correlationship of Different Indices Used (Accuracy: Upto four decimal figures).

Correlationship							
Correlations	inet rank	Freeman CC	Degree C	Betweenness	Degree C	Average hops	Hops CC
inet rank	1						
Freeman's CC	0.50013	1					
Degree	0.1033	- 0.0718	1				
Betweenness	- 0.0794	**- 0.0605**	**0.94055**	1			
Degree C	- 0.1033	- 0.0718	1	**0.9405**5	1		
Average hops	0.65892	0.52873	- 0.1600	- 0.1278	- 0.1600	1	
Hops CC	- 0.6840	- 0.5272	0.22404	0.18195	0.22404	- 0.9829	1

1. Path cost : Sum of the link costs along the path

2. Path hops : Number of hops between nodes

3. AS load : Load yielded by the packet forwarding process on each AS. It is considered proportional to the number of the packets forwarded

4.2.2 Results

The correlationships observed between mathematical indices used in determining the fit of each node to play the role of MME is give in Table 4.1. According to the results, betweenness centrality and the degree shares a correlationship of 0.9405. Further, Freeman's closeness centrality does not show any extreme correlationship with any of the other selected indices. Average hops also does not show any close correspondence with any other index. However, average number of hops is not considered much commonly in policy determination in reality. Therefore, Freeman's closeness centrality and betweenness centrality can be seen as the unique candidate properties to determine the locations of MMEs. However, calculating the betweenness closeness is extremely expensive. Thus, degree can be suggested as an approximation for betweenness centrality to be used in future studies.

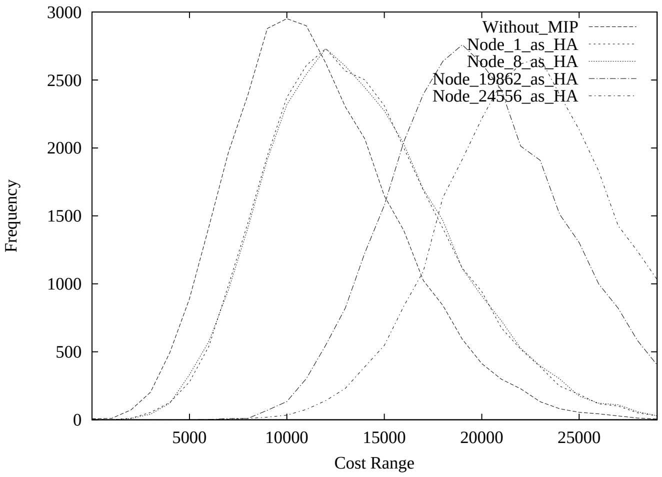


Figure 4.1: Single HA placement - Random Selection from the Whole Topology.

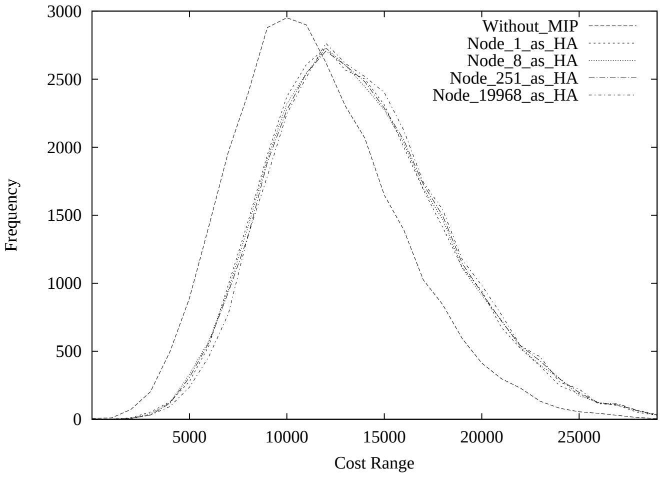


Figure 4.2: Single HA Placement - Random Selection within Nodes Holding Higher Freeman's Closeness Index.

4.2.2.1 Single HA Placement in MIPv6

Figures 4.1 and 4.2 show the impact in terms of cost, due to the single HA placement in the MIPv6 environment. Figure 4.1 can be seen as a referential evaluation as to see how the introduction of an HA affects the unmediated communication. The results emphasize that the introduction of the HA degrades the overall output of the network. Figure 4.2 shows the performance of the network when the HA is selected out of the AS nodes which maintain a higher Freeman's closeness index. This implies that the Freeman's closeness index based HA selection improves the results of single HA placement. Further, the skewness shows a direct correspondence with Freeman's closeness index.

Freeman's closeness bares a considerably preferable description of impacts generated by a single HA, compared to betweenness or any other examined measurements. As per betweenness, it is accompanied with high time consumption, high space consumption, low re-usability and hardness of extracting information afterwards. On the other hand, since it is solely based on the shortest paths between nodes which pass through the considered node, it does not include the situation where a node lies near to another, which can access it with a lower outlay though it is not travelled through in any of the shortest paths. Hence, betweenness provides fairly descriptive, but not totally accurate explanation of different vantage points when criterion concentrates on cost factor. At the same time, since Freeman's closeness is based on the minimum cost between nodes, though the node does not lie between another couple of nodes in the shortest path, if it bares a fairly small outlay, it has the opportunity to bare a least value. This justifies the results gained during the performance evaluation based on betweenness and Freeman's closeness index.

Comparison based on cost bares a decisive outcome favoring Freeman's closeness index. It is expected as Freeman's closeness index itself is calculated utilizing the average cost. In contrary, betweenness considers a node's appearance in all shortest paths in the topology. Therefore, it is anticipated to produce better results in AS load performance. Nevertheless, AS load performance between betweenness based and Freeman's closeness index based categorizations is proved to yield very close performances. Therefore, the results lead Freeman's closeness index to be regarded as the index which holds the best correspondence with single HA performance.

4.2.2.2 Multiple HA Placement in MIPv6

The graphs from Figs.4.3 to 4.10 summarize performances of multiple HA placement on a group basis. Two sets of groups are defined using two basic criteria: one is based on betweenness and the other is based on Freeman's closeness index. Each of the criteria has three groups. Importantly, this grouping mechanism uses values and the ranks of the subjected nodes in each index space separately. First, the group boundaries are determined considering the skewness of individual performances. First two groups are defined to have approximately the same value range and the final group containing the rest of the nodes. This final group includes 20,000 nodes which is 2/3 of the total number of nodes. It results from almost 2/3 of the AS nodes were earlier identified to have degrees one and two. Therefore, the assumption is that the contribution of those nodes will be less in mediation for optimality.

Using this criterion, separations of nodes are performed considering their ranks into higher, medium, and lower indices. Later, resulting graphs of five vantage points from each group are selected with equal interleaving within a group so that each selection represents the whole group.

Each graph exhibits performance of different group combinations plotted based on node ranks. Therefore, every line in the graph corresponds to a combination of two different ranks. These affiliated ranks and groups are inserted in the legend to represent each plotted line. Figures 4.3 and 4.4 show the best resulting graphs of combinations which are selected from different group combination based on betweenness and Freeman's closeness index, respectively. These different combinations are listed below.

1. Within First Group: A group 1 (G1) node is combined with another G1 node (Fig. 4.5).

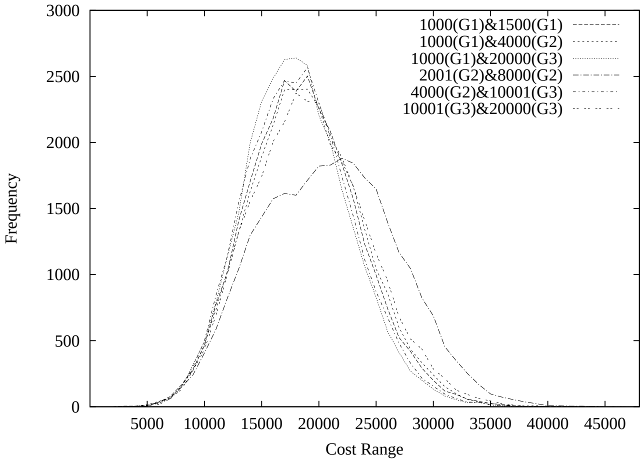


Figure 4.3: Dual HA Placement - Freeman's Closeness Index Based: Best Cases.

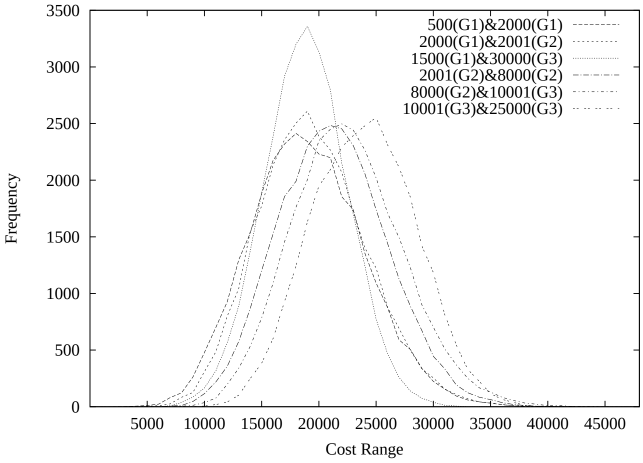


Figure 4.4: Dual HA Placement - Betweenness Based: Best Cases.

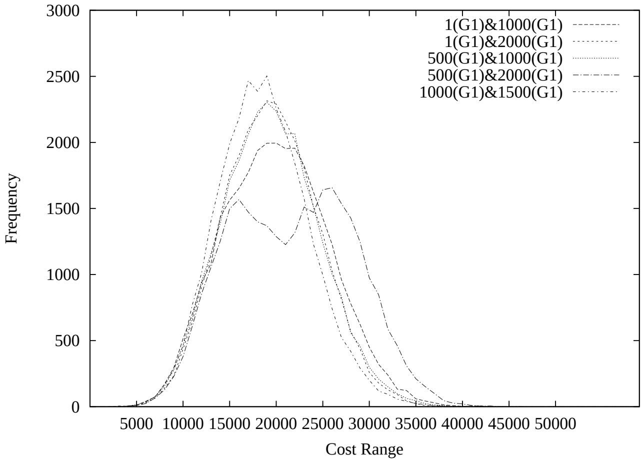


Figure 4.5: Dual HA Placement - Freeman's Closeness Index Based: Within group one.

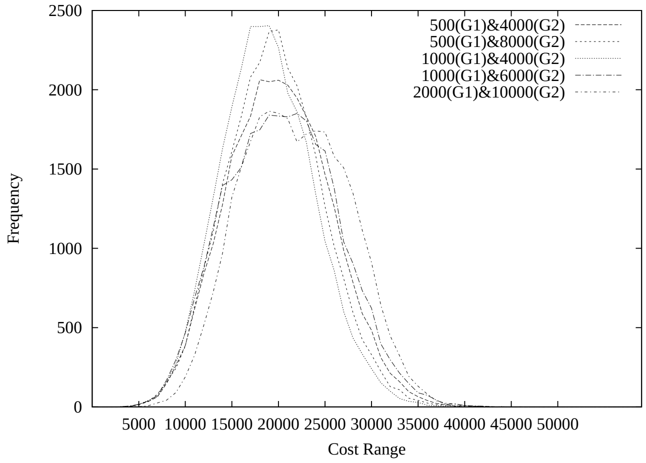


Figure 4.6: Dual HA Placement - Freeman's Closeness Index Based: group one with group two.

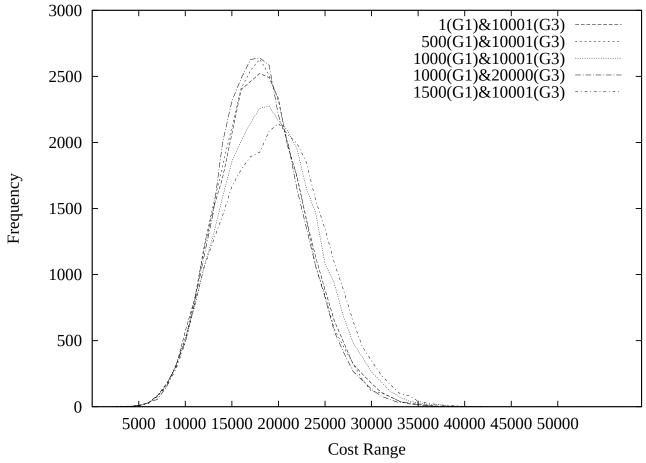


Figure 4.7: Dual HA Placement - Freeman's Closeness Index Based: group one with group three.

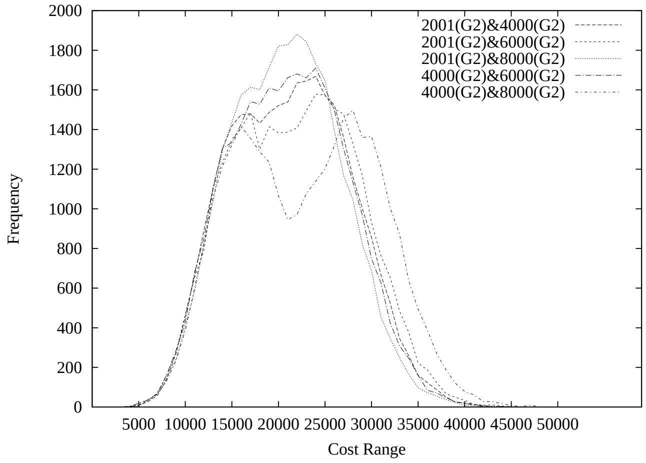


Figure 4.8: Dual HA Placement - Freeman's Closeness Index Based: Within group two.

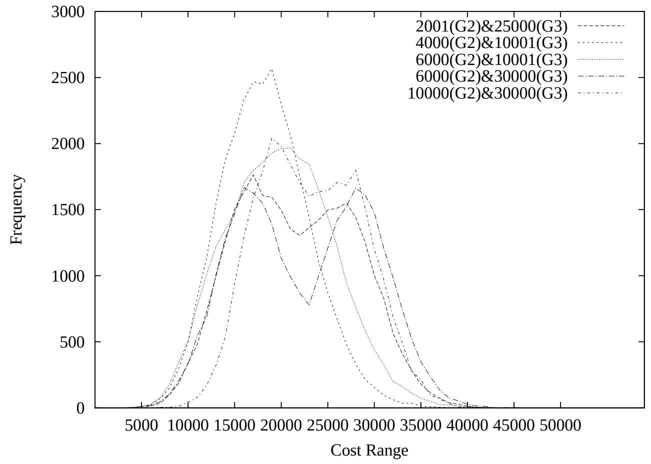


Figure 4.9: Dual HA Placement - Freeman's Closeness Index Based: group two with group three.

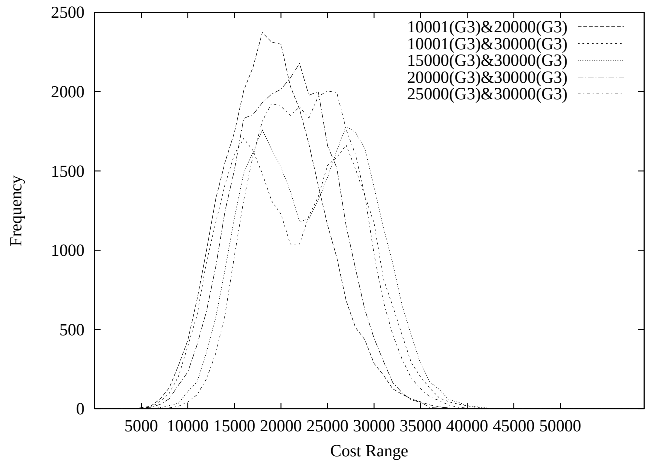


Figure 4.10: Dual HA Placement - Freeman's Closeness Index Based: within group three.

2. group one with group two: A G1 node is combined with a group two (G2) node (Fig. 4.6).

3. group one with group three: A G1 node is combined with a group three (G3) node (Fig. 4.7).

4. Within group two: A G2 node is combined with another G2 node (Fig. 4.8).

5. group two with group three: A G2 node is combined with a G3 node (Fig. 4.9).

6. Within group three: A G3 node is combined with another G3 node (Fig. 4.10).

Results show that the Freeman's closeness index based HA selection improves overall performance of the data-plane. Thus, it can be considered as an appropriate determinator for HA placement. Then next set of graphs, Figs. 4.5 to 4.10, shows the performance of dual HA placement for different combinations of HA.

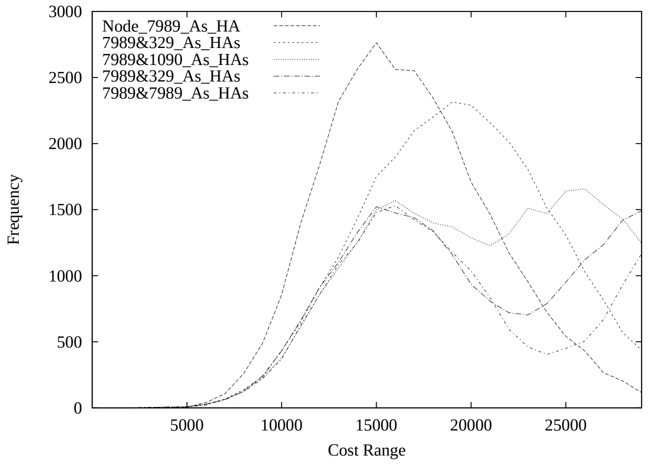


Figure 4.11: Freemans Closeness Based: Single HA vs Multiple HA Placement Within group one.

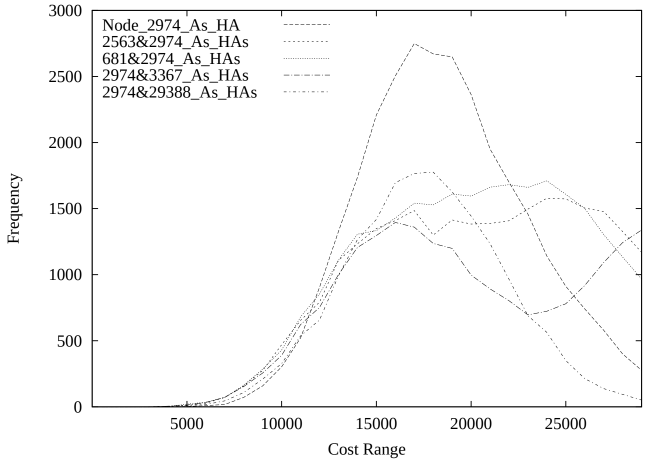


Figure 4.12: Freemans Closeness Based: Single HA vs Multiple HA Placement Within group two.

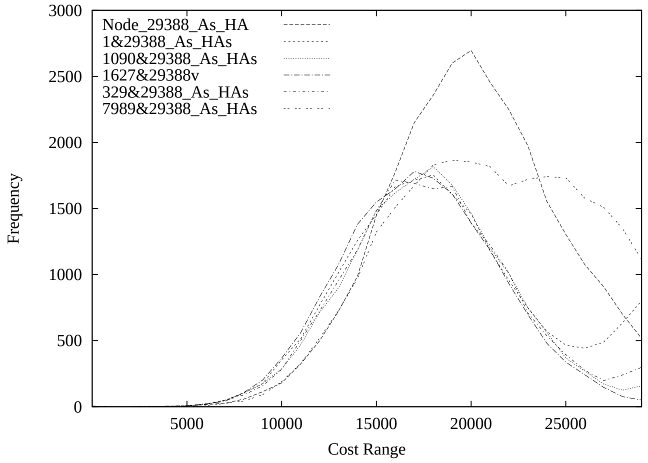


Figure 4.13: Freemans Closeness Based: Single HA vs Multiple HA Placement of a group two Combined with group one.

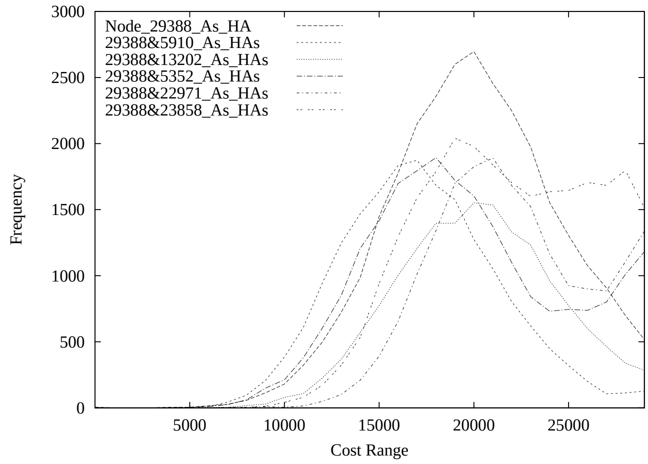


Figure 4.14: Freemans Closeness Based: Single HA vs Multiple HA Placement of a group two Combined with group three.

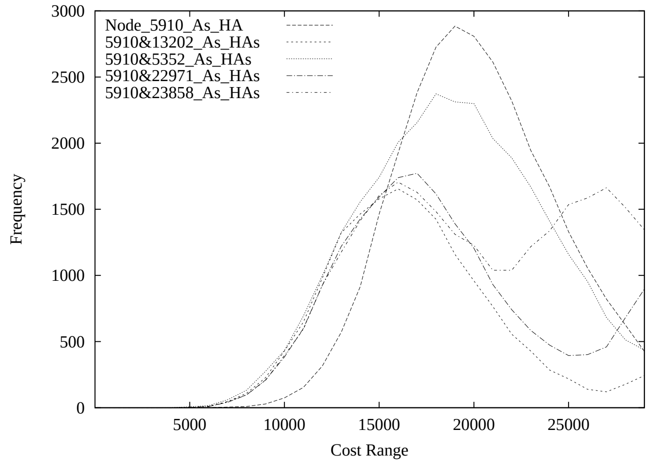


Figure 4.15: Freemans Closeness Based: Single HA vs Multiple HA Placement Within group three.

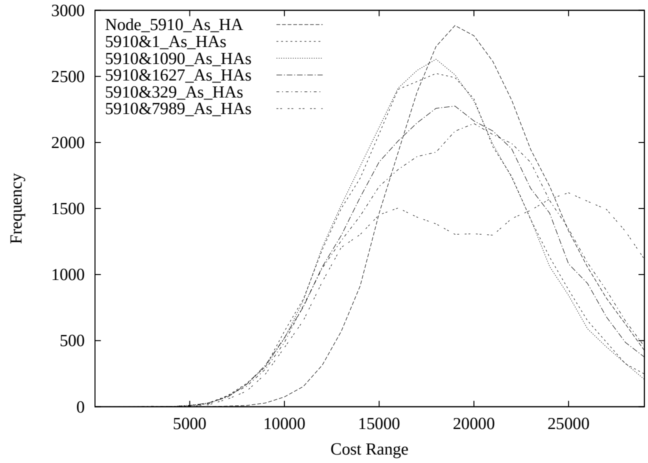


Figure 4.16: Freemans Closeness Based: Single HA vs Multiple HA Placement of a group three Combined with group one.

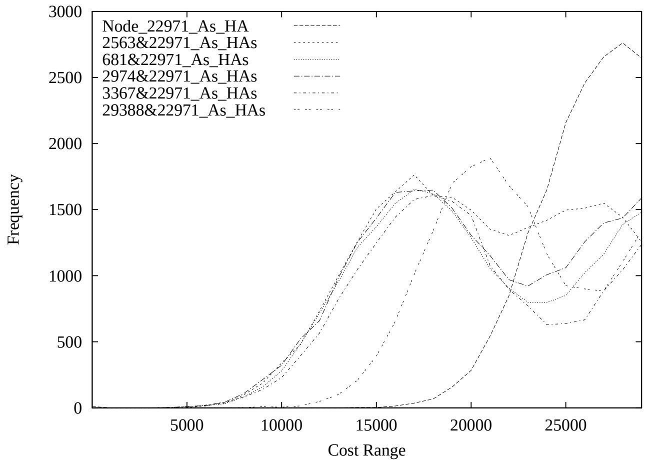


Figure 4.17: Freemans Closeness Based: Single HA vs Multiple HA Placement of a group three Combined with group two.

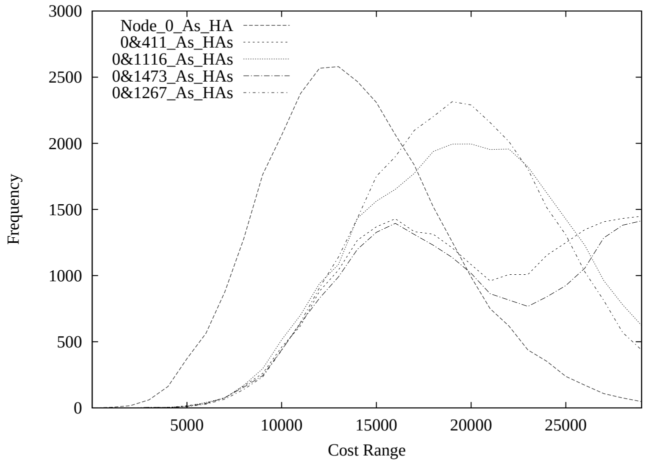


Figure 4.18: Betweenness Based: Single HA vs Multiple HA Placement Within group one.

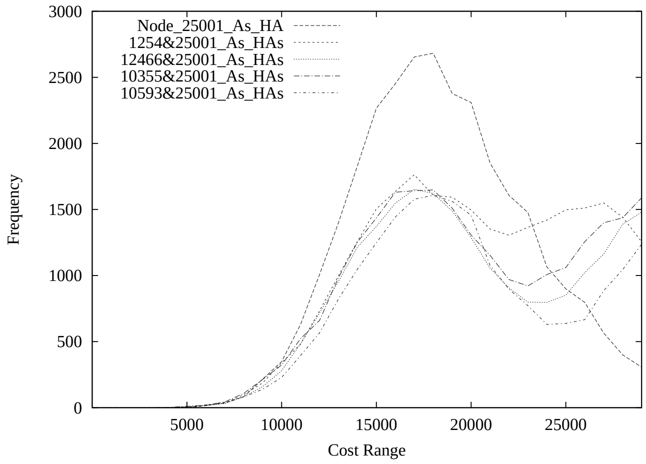


Figure 4.19: Betweenness Based: Single HA vs Multiple HA Placement of a group three Combined with group one.

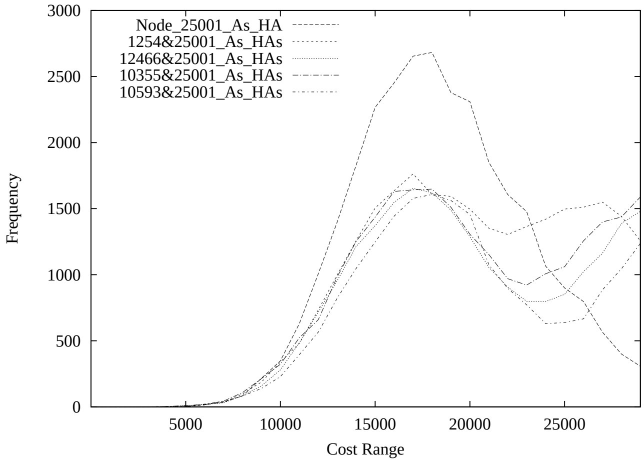


Figure 4.20: Betweenness Based: Single HA vs Multiple HA Placement of a group three Combined with group two.

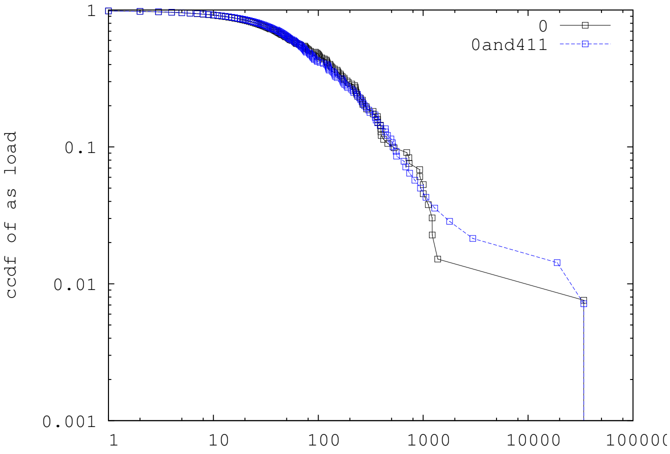


Figure 4.21: Betweenness Based: AS Load of Single vs Multiple HA of a Random Combination.

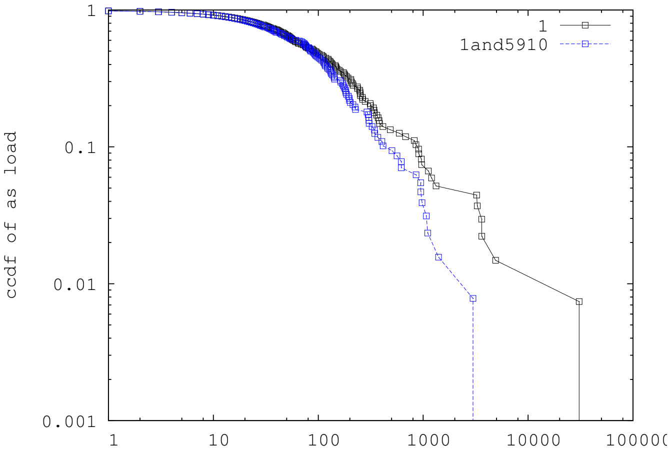


Figure 4.22: Freeman's Closeness Index Based: AS Load of Single vs Multiple HA of a Random Combination.

Table 4.2: New Grouping of AS Nodes.

Group	1	2	3	4	5
Range of links	over 500	50-499	10-49	3-9	less than 2
Number of nodes	13	175	928	6,197	23,429

Figures 4.18 to 4.20, show the performance of dual HA placement for different combinations of HA. Finally, Figs. 4.21 to 4.22 show the AS loads for single and multiple Ha placements during the betweenness and Freeman's closeness based HA placements, respectively.

4.3 Performance Evaluation of Host-based DMM Schemes in the Internet

This section provides a detailed description of the evaluation that was carried out in order to identify the best practice to achieve DMM. Since the best HA placement is distinguished by the earlier evaluation in Sect. 4.2, the results obtained are used in this evaluation to install multiple HAs in the MIPv6 environment.

4.3.1 Approach

MIPv6 (Sect.2.3) with indirect routing (Sect.2.3.1) with RR (Sect.2.3.2) are used as the referential centralized schems. Then, Global HAHA (Sect.2.4), Migrating HA (Sect.2.5), and Distributed MIPv6 (Sect.2.7) are used as the candidate schemes for the evaluation. As far as DMIPv6 is concerned, both alternative approaches that are discussed in Sect.2.7 are considered.

Assuming the global scale and host based mobility, this evaluation attempts to investigate the performance of the above schemes using the real Internet AS-level data. Thus, the mapped AS-level topology of the Internet is obtained from the CAIDA Internet data repository (Sect. 3.4.1.2). The *AS relationship* data set is used, which is stored under the topology category [13]. Unlike the synthetic AS-level topology of the Internet, this data gives an insight to the policy based routing. Thus, the relationship information and the roles that are played by the ASes are fed to the simulator described in Sect. 3.5.2.

Further, in order to intensify the accuracy of the results, tier-2 and tier-3 are further split. This is due to the number of the ASes in each of these groups are larger. Thus, new classification contains five groups. Tier-1 is considered as group 1. Then, tier-2 is divided into two groups, group 2 and group 3. Tier-3 is also divided into two groups as group 4 and group 5. The contents of the new groups are shown in Table 4.2

Figure 4.23 highlights the basic setup of the evaluation scenario. Starting from the generic MIP and MIPRR, considering the briefed decentralized MM schemes and placing HAs at different candidate AS locations, communications between random MNs and CNs are evaluated as described below.

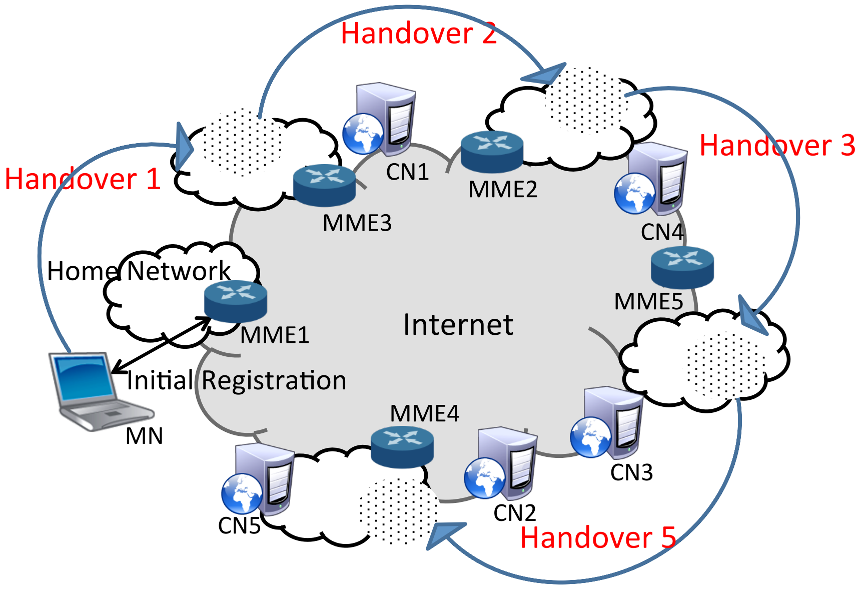


Figure 4.23: Evaluation Scenario.

Two random AS node sets of 5 × 10,000 CN locations and 5 × 10,000 MN locations are generated (10,000 trials). Each trial represents five different locations the MN occupies, starting from the home/default network. At the initial location, the MN starts a session with *CN1*. At each location the MN occupies after consequent handovers, it starts a new session with the respective CN in the CN data set. For example, according to Fig. 4.23, after *Handover 1*, the MN starts a new session with *CN2* while keeping its session with *CN1* alive. After *Handover4* takes place, the MN will have five different running sessions with *CN1* through *CN 5*. This trial scenario imitates frequent handovers with shorter residence time which we assume to be the most critical in terms of cost. At each location MN is assumed to start a new session which it tries to continue while it keeps alive the already started sessions. 10,000 such scenarios are examined in order to draw conclusions.

4.3.2 Mathematical Model

In order to analyze the performance of each of the selected DMM schemes, it is required to model the control and data-planes of each scheme as a sub-emulator of the DMM scheme emulator discussed in Sect. 3.11. Thus, MIPv6 Sect.2.3 with indirect routing Sect.2.3.1 with RR Sect.2.3.2, Global HAHA (Sect.2.4), Migrating HA (Sect.2.5), and Disributed MIPv6 (Sect.2.7) are mathematically modeled. Following gives the main models that are derived for the control and data-planes.

lET $Cost_C$ be the additional cost incurred in the control plane due to the handover $Cost_c$. This cost can be calculated in terms of routing cost, $1/2$ × Round Trip Time (RTT) or the total number of hops. The communication cost in the resultant data plane is symbolized as $Cost_d$. Similar to the control-plane cost, it also can be calculated based on the same measurements.

Let n be the number of CNs with which the MN is trying to keep communication and let m be the number of MMEs (degree of distribution). Let $C_{x,y}$ be the one way communication cost between entity *x* and entity *y*, where the entity *x* and the entity *y* can be either an AS node to which *CN* or *MN* belongs or an AS maintaining an MME such as *HA*, *LM*, or *MR*. In situations where there are multiple entities of CNs or MMEs, *i* is used to distinguish among them as CN_i, HA_i, LM_i, or MR_i. Here, *i* is bounded by *n* in case of CN or *m* in case of HA, LM, or MR. In cases of Global HAHA and Migrating HA, let HA_k be the HA closest to *k*, where *k* can be CN or MN and the HA_p be the previous HA of MN. In DMIP, let X_h be the MMEs in the home network, where *X* can be LM or MR. The HoA function is assumed to co-locate with the MR.

4.3.2.1 Mobile IPv6

MIPv6 control-plane for indirect routing can be given by the formula

$$Cost_c = 2 \times \sum_{i=1}^{n} C_{mn,ha}. \tag{4.1}$$

MIPv6 data-plane for indirect routing,

$$Cost_d = C_{cn,ha} + C_{mn,ha}. \tag{4.2}$$

4.3.2.2 Mobile IPv6 with Return Routability Procedure

MIPv6-RR control-plane, which supports direct routing can be given by the formula

$$Cost_c = 2 \times C_{mn,ha} + \sum_{i=1}^{n} C_{mn,ha} + \sum_{i=1}^{n} C_{ha,cn_i} + 3 \times \sum_{i=1}^{n} C_{cn_i,mn}. \tag{4.3}$$

MIPv6 data-plane for indirect routing can be given by the formula

$$Cost_d = C_{cn,mn}. \tag{4.4}$$

4.3.2.3 Global HAHA

The control-plane cost of the Global HAHA scheme can be given by the formula

$$Cost_c = 2 \times C_{mn,ha_{mn}} + C_{ha_{mn},ha_p} + C_{ha_p,mn} + \sum_{i=1}^{n-1} C_{ha_{mn},ha_{cn_i}}. \tag{4.5}$$

The data-plane cost of the Global HAHA can be given by the formula

$$Cost_d = C_{cn,h_{cn}} + C_{mn,ha}. \tag{4.6}$$

4.3.2.4 Migrating HA

The control-plane cost of the Migrating HA scheme can be given by the formula

$$Cost_c = 2 \times C_{ha_{mn},mn} + 2 \times \sum_{i=1}^{n-1} C_{ha_{mn}ha_i}. \tag{4.7}$$

The data-plane cost of the Migrating HA can be given by the formula

$$Cost_d = C_{cn,h_{cn}} + C_{mn,ha}. \tag{4.8}$$

4.3.2.5 Distributed MIPv6 (common)

The common control-plane cost of the DMIPv6 scheme can be given by the formula

$$Cost_c = 2 \times C_{mn,mr_{mn}} + C_{mn,mr_{mn}} + C_{mr_{mn},lm_h} + C_{lm_h,mn}. \tag{4.9}$$

4.3.2.6 Distributed MIPv6 (1st Approach)

The additional control-plane cost of the 1st approach discussed under DMIPv6 scheme can be given by the formula

$$Cost_c = 2 \times C_{lm_h,mr_h} + C_{lm_h,mr_{cn}}. \tag{4.10}$$

The data-plane cost of the 1st approach discussed in DMIPv6 can be given by the formula

$$Cost_d = C_{cn,mr_{cn}} + C_{mr_{cn},mr_h} + C_{mr_h,mr_{mn}} + C_{mr_{mn},mn}. \tag{4.11}$$

4.3.2.7 Distributed MIPv6 (2nd Approach)

The additional control-plane cost of the 2nd approach discussed under DMIPv6 scheme can be given by the formula

$$Cost_c = 3 \times C_{lm_h,mr_{cn}}. \tag{4.12}$$

The data-plane cost of the 2nd approach discussed in DMIPv6 can be given by the formula

$$Cost_d = C_{cn,mr_{cn}} + C_{mr_{cn},mr_{mn}} + C_{mr_{mn},mn}. \tag{4.13}$$

4.3.3 Results

4.3.3.1 Total Number of Hops

Figures 4.24 and 4.25 show CCDFs of the best performances yielded during the analysis, having the MME combination of the groups 1 and 4, at the control plane and the data plane, respectively. Figures 4.26 and 4.27 illustrate the worst performances, which were observed for the MME combination within the group 5.

MIP exhibits the best control-plane performance. In contrast, it yields the worst results in the data-plane.

On the other hand, MIPRR shows the best-data plane performance, as a result of achieving routing optimization. However, it in turn exhibits the worst control-place performance. This is as a result of the control-plane overhead it introduces due to the return routability procedure. Thus, it can be seen extremely expensive in the control-plane, when compared with all the other considered DMM schemes.

There can be seen a general performance pattern in the control-plane for the best case and for the worst case. Nonetheless, MIP seems to show degraded results during the worst case, where the other schemes seems to maintain the same cost. Thus, it can be seen that Migrating HA, Global HAHA, and the two DMIP schemes seem to have a stable control-plane, despite of the HA installment.

When the data-plane performance is compared, MIPv6 without RR and MIPRR show the marginal performance for both best and worst cases. All the other schemes,

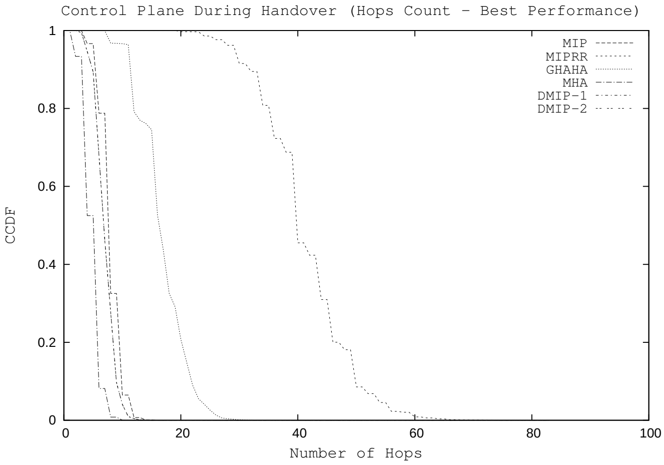


Figure 4.24: Handover cost in control plane in terms of number of hop: best case (Combination of groups 1 and 4).

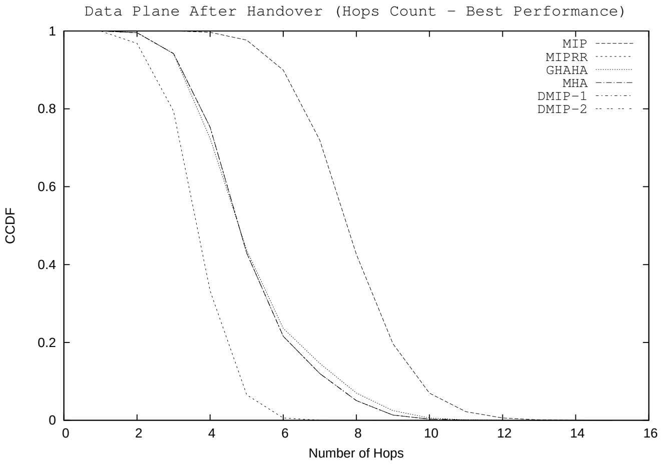


Figure 4.25: Number of hops in the data plane: best case (Combination of groups 1 and 4).

i.e, Migrating HA, Global HAHA, and the two DMIP schemes lie in between leaned towards MIPRR in the best case. In contrast, they are leaned towards MIPv6 in during the worst care. This indicates performance degradation in overall.

4.3.3.2 AS Load

CDFs of the AS load in control and data-planes are depicted in Figs. 4.28 and 4.29, respectively. Note that an inverse x-axis is used in the above mentioned graphs (Figs. 4.28 and 4.29).

When the control-plane AS load is considered, the result seems to cluster the MM schemes into three groups with clear boundaries. MIPRR shows the worst results in the control-plane (Fig.4.28), which seems to introduce an excessive level of AS load within the network. Migrating HA and the two DMIP schemes display the least AS load. MIP and Global HAHA fall in between marking three different performance groups.

When the obtained results are considered, the data-plane performance in terms of AS load trend to be similar. There cannot be seen any significant difference. However, one of the DMIP schemes and MIPRR result in a slightly better traffic condition in the

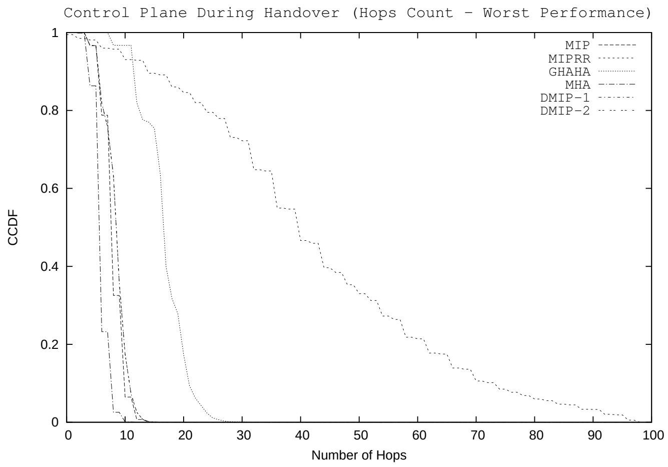


Figure 4.26: Handover cost in control plane in terms of number of hop: worst case (Combination within group 5).

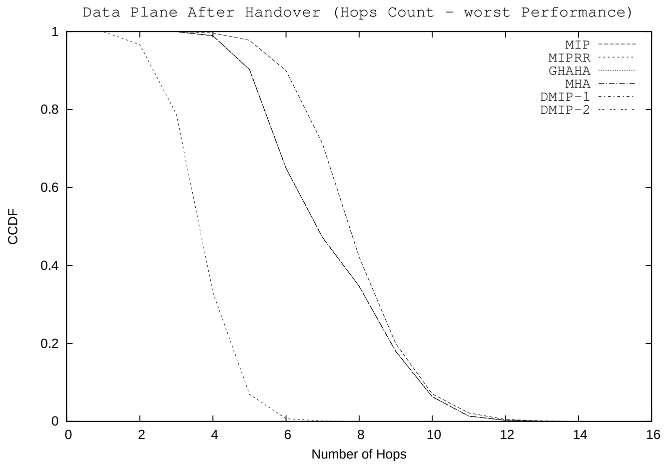


Figure 4.27: Number of hops in the data plane: worst case (Combination within group 5).

data-plane than the rest of the schemes.

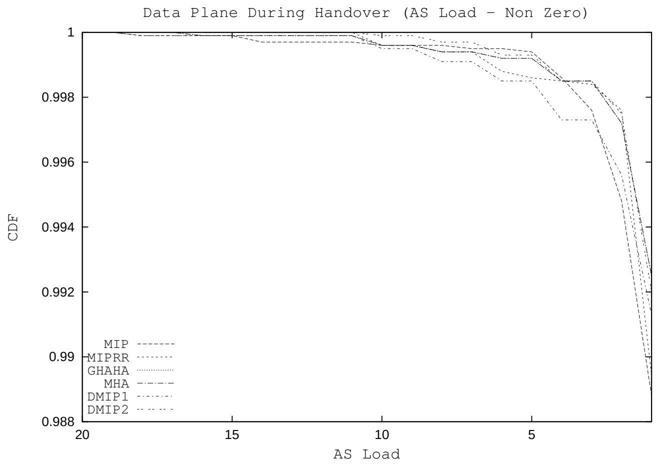


Figure 4.29: Cost in data plane in terms of AS load.

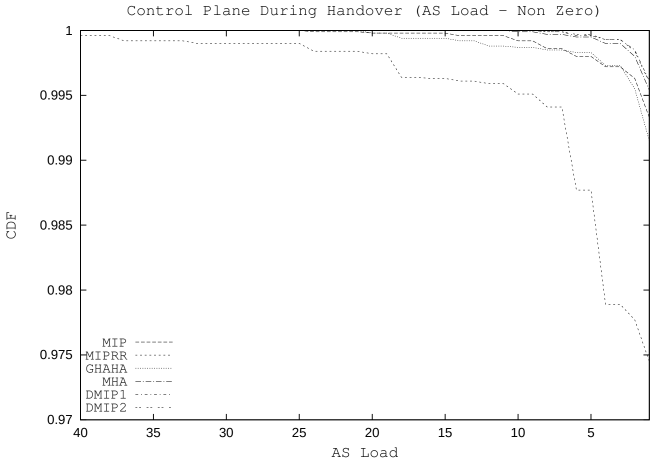


Figure 4.28: Handover cost in control plane in terms of AS load.

4.3.4 Overall Findings

As an overall impression, it can be suggested that MIPRR has the best data-plane routing regardless of MME placement. However, in order to obtain routing optimization, it produces an extremely large control-plane overhead. The RR procedure is immensely complex and extremely limited in scalability, which makes it unrealistic. Especially it cannot be reccommended for Internet scale. Because, revealing the location of the MN to external nodes might introduce security vulnabilities. Comparatively, Global HAHA improves MIP, but it has excessive traffic in the control-plane which is further improved by Migrating HAHA. Still, they require a consistency mechanism which would be costly with the scale of distribution. At the same time, both the DMIP schemes produce the same impact on the data-plane and negligible negative performance deviation on the control-plane with the advantage of absolved consistency maintenance and enriched scalability.

When the AS load is concerned, it does not show much significance. That emphasizes that, when the topology is concerned as a whole, the suggested control and data-planes of the selected schemes do not differ in terms of the traffic they excert on the topology as a whole. MIPRR, Migrating HA, and one of the DMIP schemes outperform the rest, whereas MIP maintains the highest impact on data traffic. The second DMIP scheme requires data rerouting identical to MIP in the MN's home network when it routes the first packet, which decreases its performance slightly. For consecutive data forwarding, it attains the same state as the first DMIP scheme. In contrast, the second DMIP scheme has a slightly increased control-plane traffic when forwarding the first packet, which has affected its control-plane performance.

4.4 Summary

This chapter introduced a set of evaluations that were carried out envisioning to determine the best practice to provide Distributed Mobility Management (DMM) in the global scope. Target environment is a MIPv6 equipped IPv6 environment.

The only MM Entity (MME) employed in the IPv6 environment is the Home Agent

(HA). First, it is identified that the HA placement in the Internet should be optimized to obtain feasible results with candidate schemes. Thus, the first evaluation looks at the multiple HA installment in a MIPv6 environment. Global HAHA and Migrating HA were considered as candidate schemes apart from MIPv6.

Fitness of the nodes to play the role of HA was examined. A several different indices like, degree, Freeman's closeness, betweenness centrality and average hops were used as fitness parameters. Betweenness centrality was found closely related to the degree centrality. Calculating betweenness centrality bares a high computational cost. Thus, it was recommended to be approximated by degree centrality. Then, the shortest path based Freeman's closeness index and average hops were also recognized as a unique indices. However, in reality, average hops is not considered as a strong measurement. Thus, it was not used.

Considering the three tier categorization of Autonomous Systems (AS) in the Internet, a synthetic AS-level topology of 30,000 nodes was grouped into three groups based on the indices betweenness and Freeman's closeness centralities. Then, the single and dual HA placements were evaluated for different HA combination of candidate nodes selected from the three groups. Results show that the Freeman's closeness centrality yields a better classification of nodes. Thus, the groups behave different with different HA installments. However, groups based on betweenness centrality fails in show unique behavior.

Secondary HA installment shows that in neither case a selection of both HAs from the first group gives the best performance. Focusing on Freeman's closeness based categorization, it suggests single HA placement is best for the first group. For the second group, the upper part yields the same result as the first group and if an additional installment needed, it would be for the latter part and should be coupled with an HA belonging to the third group. For the third group, the performance can be improved by using a collaboration with the first group.

Using the results obtained during the above evaluation, the next evaluation compares the host-based DMM schemes found in the literature. Unlike the previous data set, a real AS-level topology data obtained with AS relationships was used during comparison evaluation. Thus, realistic routing policies were also considered. MIPv6, MIPv6 with Return Routability (MIPRR), Global HAHA, Migrating HA, and Distributed MIPv6 (DMIP) were considered as candidate schemes. Global HAHA and Migrating HA suggest Mobility Management Entity (MME) duplication. In contrast, DMIP simplifies requirements per MME into Home Address (HA) assignment, Location Management (LM), and Mobility Routing (MR). Then, distributes those functions separately.

Comparisons made on the total number of hops and the AS load traces on both the control and data-planes show contrasting results. Though MIPRR results in better data-plane routing, it bares a speculative signaling procedure. Migrating HA outperforms Global HAHA, but it fails to overcome the opportunity cost of consistency maintenance and limited scalability. In contrast,the two DMIP schemes were competent of maintaining better performances overruling the consistency and scalability problems.

Thus, it can be concluded that a fully distributed MM scheme (i.e, distributed in both the control and data planes) like DMIP, exhibits sensible efficiency while mitigating the problems the legacy centralized and distributed schemes have. However, it introduces an additional cost of intercommunication between different mobility related functioanlities.

Chapter 5

Evaluation of Localized Distributed Mobility Management Schemes

Localization of the MM suggests improvements to the idea of host-based MM by cutting off the MM signalling between the MN and the network. It also has the advantages of capability to facilitate unmodified MNs. Mobility of MNs within the local domain can be considered as the most common mobility scenario. Thus, it can be seen as an appropriate approach to handle mobility.

The rest of the chapter introduces the evaluation which is carried out in order to identify the best approach to facilitate DMM within the local domain.

5.1 Motivation

Literature lacks evaluation studies of DMM schemes in the local-domain. Lack of simulator support to analyzed DMM concepts is seen as one of the major disadvantages which restricts evaluation. Further, there is no standardization for the MM in the local domain. Evaluation can be considered as a major contribution towards standardization.

5.2 Performance Evaluation of Intra-AS Distributed Mobility Management Schemes

This section provides an overview of the evaluation carried out, envisioning to determine the best approach to provide localized DMM. This evaluation utilizes five different DMM schemes that fall under three main categories.

1. Fully-Distributed : Extended PMIPv6 (Sect.2.9) and DPMIPv6 (Sec.2.10)

2. Control/Data-plane Split : Address Delegation (Sect.2.11) and RO-SDN (Sect.2.12)

3. Destination-end static anchoring : Correspondent Homing (Sect.2.6)

Table 5.1 summarizes the basic properties of the selected DMM Schemes.

Table 5.1: Characteristics and Features of Selected DMM Schemes.

	Fully Distributed		Control/Data plane split		Destination-end Anchoring
	Extended PMIPv6	DPMIPv6	RO-SDN	Address Delegation	Correspondent Homing
MM Entity	MAG	D-GW	SDN-Con	AR	CHA
Anchor	Static		Dynamic		Static (CHA)
Optimal data-plane routing routing	×		●		▲
MM Information Redundancy	●		▲		●

5.3 Approach

Considering the ISP topologies discussed in Sect. 3.4.2.2, a random set of access points upto 200 ARs is selected as default residential locations of the MN. This selection is performed based on the number of ARs in the selected topology. Then, considering the selective mobility pattern generation discussed in Sect .3.7.2, five different handovers are determined for each MN. For each of these test scenarios, a set of 50 sessions is assumed. Both local communications as well as sessions in which the MN corresponds with external CNs are considered.

A simple traffic pattern is assumed in this evaluation. Sessions are established at the each location according to the generated session data set. Then data packets are sent at a constant rate triggered by the initial connection. After the detachment, the CN presumes to send packets at the same rate. Thus, the number of packets sent during the handover will be proportional to the handover time. After the successful completion of the handover, the delay and the cost for receiving the first packet would therefore be consistent with the time it takes to deliver the packets buffered by the MME if there are any. Further, packet offloading is considered for the schemes that support this mode.

5.4 Results

5.4.1 Control Plane Cost in Performing Handover

Figures 5.1-5.4 show the four CCDFs of the control plane costs observed at Sprint (ASN: 1239), Tiscali (ASN: 3257), Telstra (ASN: 1221), and Exodus (ASN: 3967) topologies, respectively. The x-axis represents the cost which is calculated based on Rocketfuel link weights.

In general, Extended PMIP scheme shows a lower cost due to the limited amount of signalling it employs in handling mobility. In contrast, Address Delegation and RO-SDN DMM schemes show the highest costs, as they require routing table updates at all the routers within the ISP. As a result, the resultant cost is above the plotted range depicted by a flat line at value 1 in the y-axis. Looking at the control plane cost in the Sprint topology (Fig.3.6), it shows degraded results for Correspondent Homing scheme, whereas it seems to perform better in all the other three topologies. The reason can be

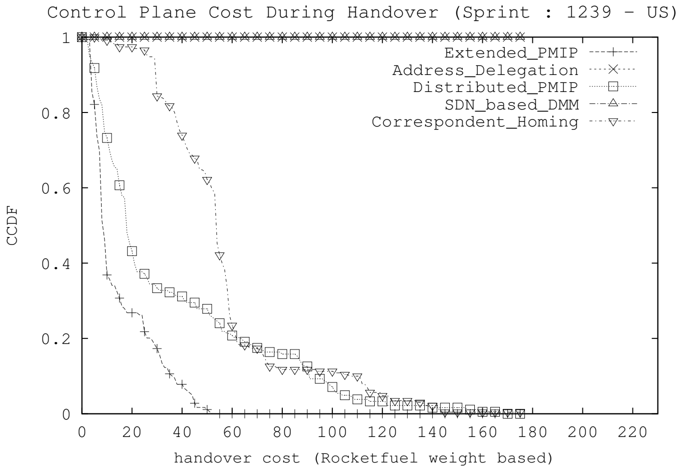


Figure 5.1: Control-plane Cost in Performing Handover : Sprint (1239).

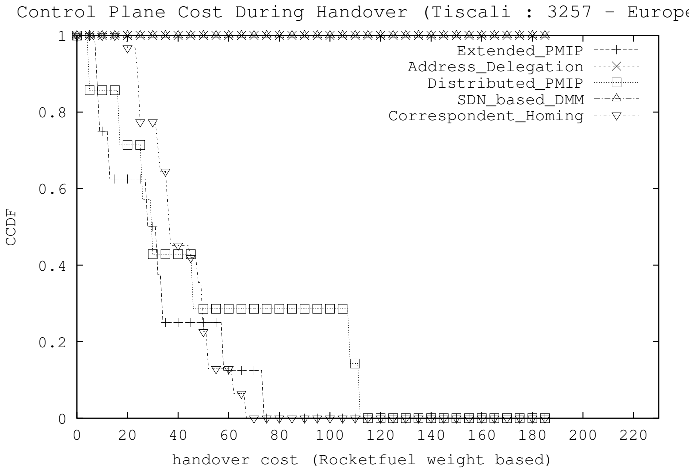


Figure 5.2: Control-plane Cost in Performing Handover : Tiscali (3257).

seen as distribution of the topology where it results in a MIP like behavior due to the centralized MME. However, in smaller topologies it seems to perform better.

5.4.2 Latency in Control Plane During Handover

The CCDFs of the control plane costs observed at Sprint, Tiscali, Telstra, and Exodus topologies are plotted in Figs.6.28-6.31, respectively. The x-axis of the graphs represents the latency encountered during handover. In Address Delegation scheme and RO-SDN schemes, the convergence time is used in calculating respective latencies. Just as in the control cost, a decline of performance of the Correspondent Homing scheme can be seen due to the same reason described in Sect.5.4.1.

5.4.3 Data Plane Cost to Deliver the First Packet

Next set of figures, Figs.5.9 through 5.12 represent the data plane costs in delivering the first data packet after the handover occurs. The x-axis denotes the data plane cost. In

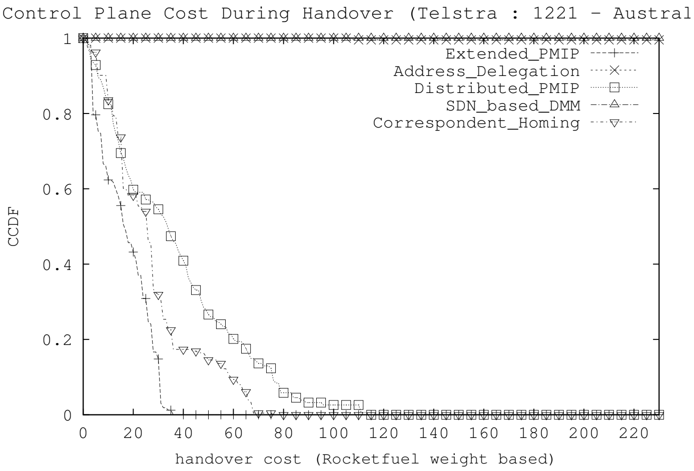


Figure 5.3: Control-plane Cost in Performing Handover : Telstra (1221).

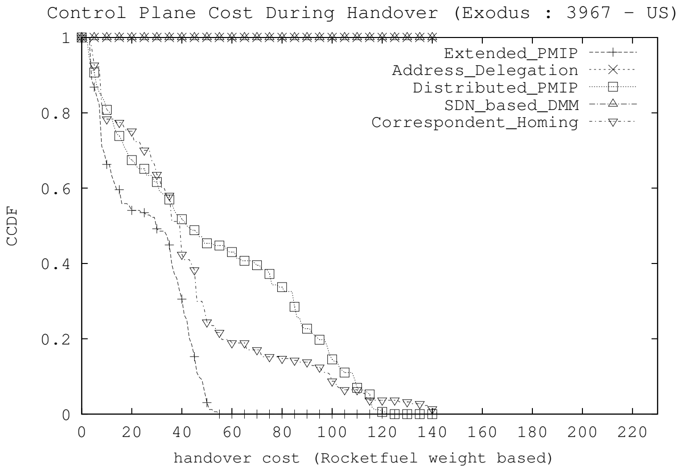


Figure 5.4: Control-plane Cost in Performing Handover : Exodus (3967).

overall, in all the topologies except for Tiscali topology, the same cost pattern for all the schemes can be observed, but with decreasing costs, as the ISP tier gets lower. Address Delegation and RO-SDN schemes show the least costs, which are theoretically optimal. Correspondent Homing scheme shows relatively poor performance as expected based on the control plane observations. Extended PMIP and Distributed PMIP schemes have intermediate results in general.

5.4.4 Data Plane Latency in Delivering the First Packet

Considering the latency in the data-plane observed in Figs.5.13-5.16, they obviously reflect the performance variances which occur due to the underlying ISP topology. Here again, the same effects encountered earlier can be visible in performance of Correspondent Homing scheme. Address Delegation and RO-SDN schemes attain the same performance level as in the router cost. The basic performance pattern is almost the same as the data-plane cost described earlier in Sect.5.4.3.

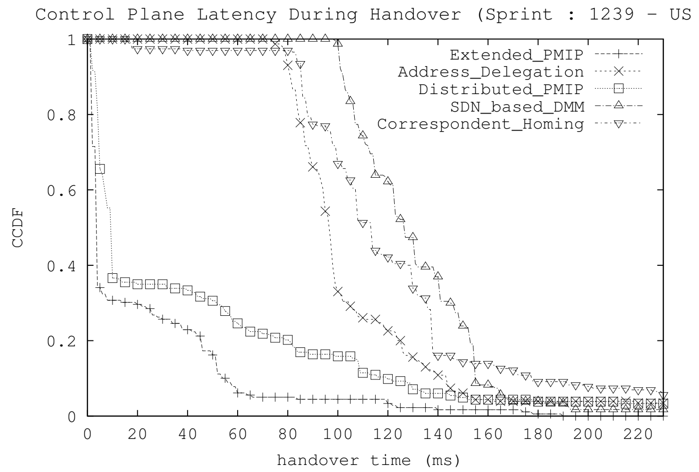


Figure 5.5: Latency in Control-plane During Handover : Sprint (1239).

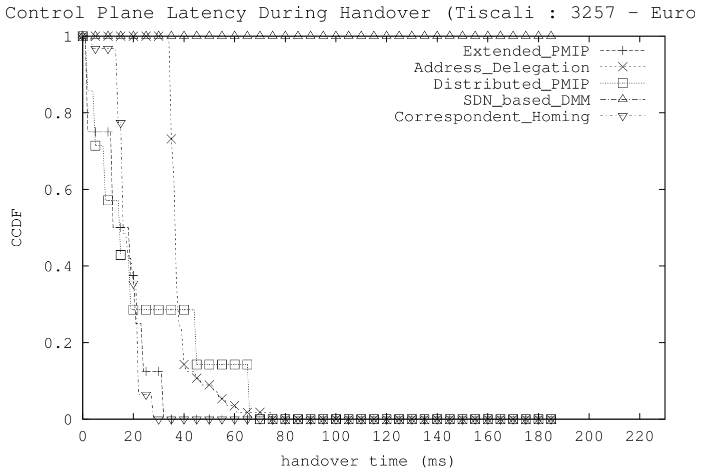


Figure 5.6: Latency in Control-plane During Handover : Tiscali (3257).

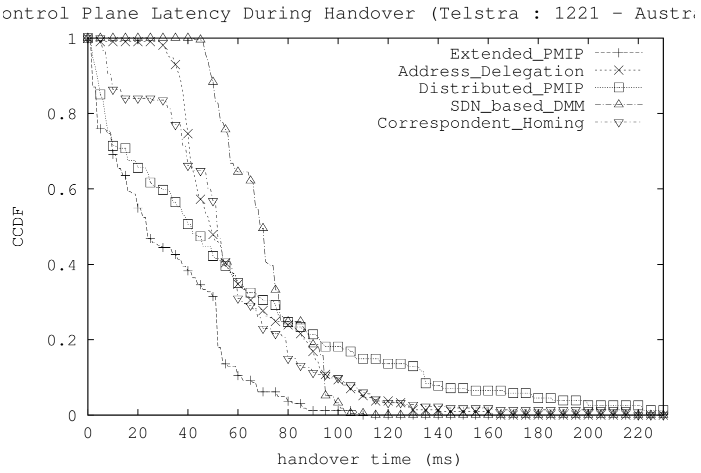


Figure 5.7: Latency in Control-plane During Handover : Telstra (1221).

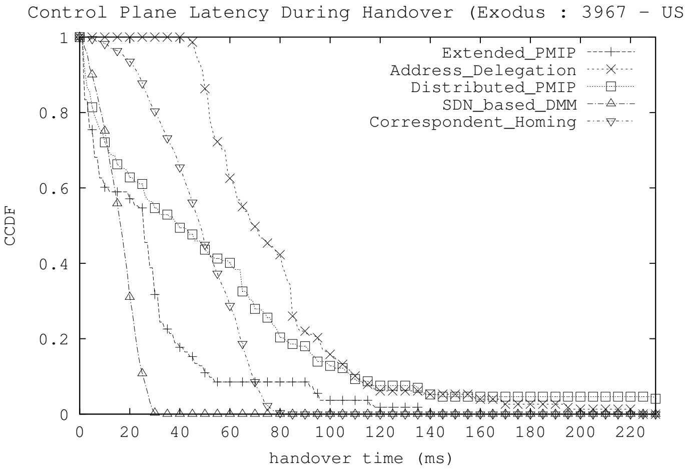


Figure 5.8: Latency in Control-plane During Handover : Exodus (3967).

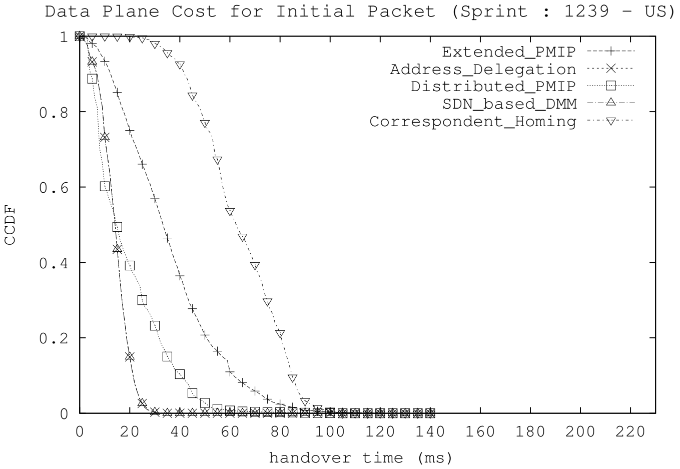


Figure 5.9: Data-plane Cost to Deliver the First Packet : Sprint (1239).

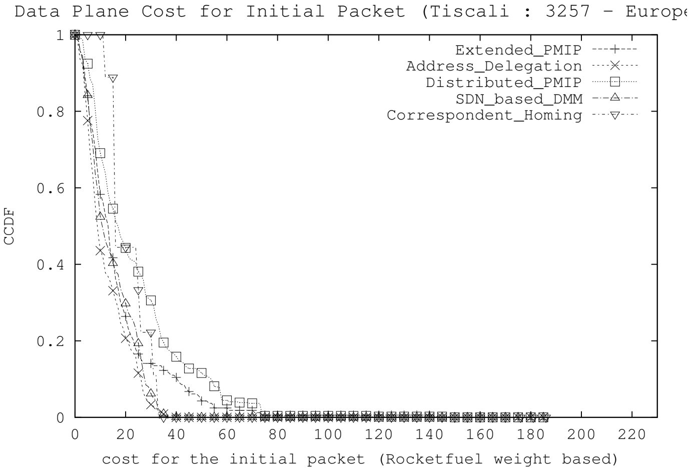


Figure 5.10: Data-plane Cost to Deliver the First Packet : Tiscali (3257).

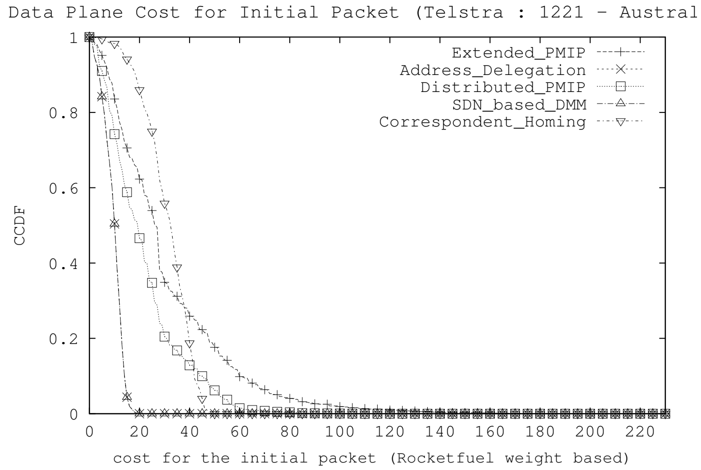


Figure 5.11: Data-plane Cost to Deliver the First Packet : Telstra (1221).

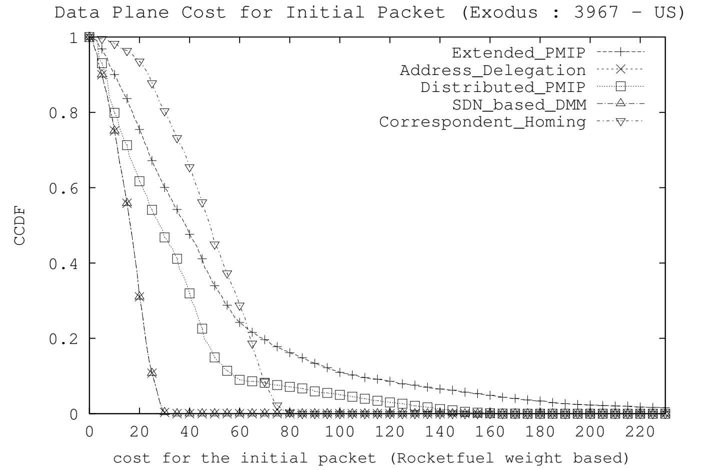


Figure 5.12: Data-plane Cost to Deliver the First Packet : Exodus (3967).

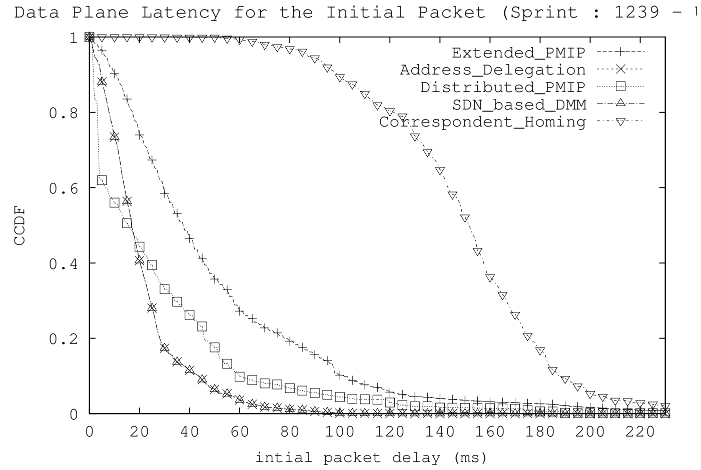


Figure 5.13: Data-plane Latency occurred in Delivering the First Packet : Sprint (1239).

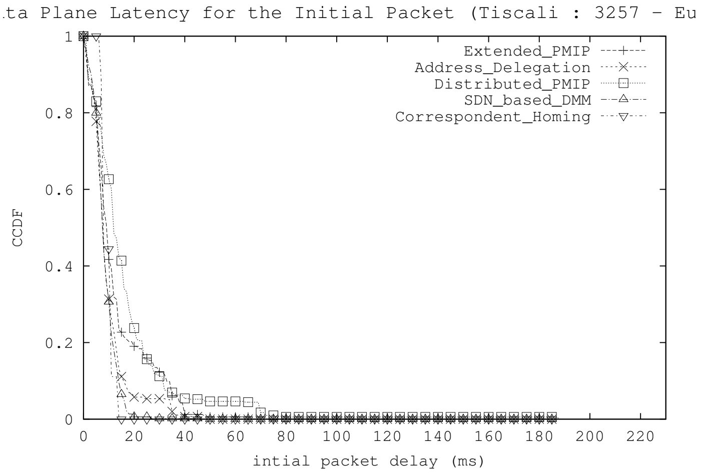


Figure 5.14: Data-plane Latency occurred in Delivering the First Packet : Tiscali (3257).

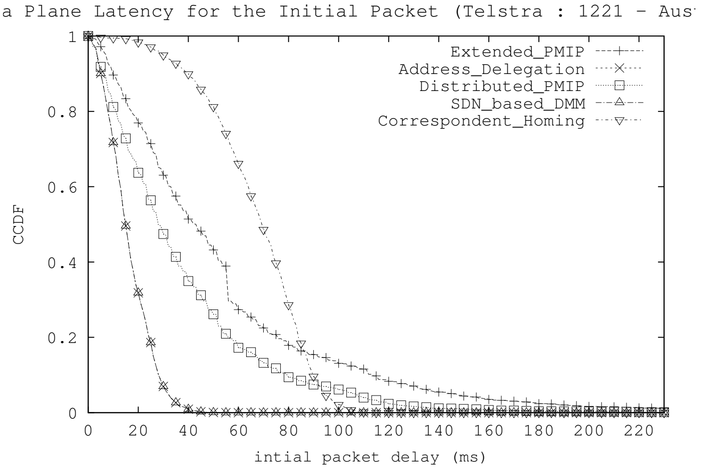


Figure 5.15: Data-plane Latency occurred in Delivering the First Packet : Telstra (1221).

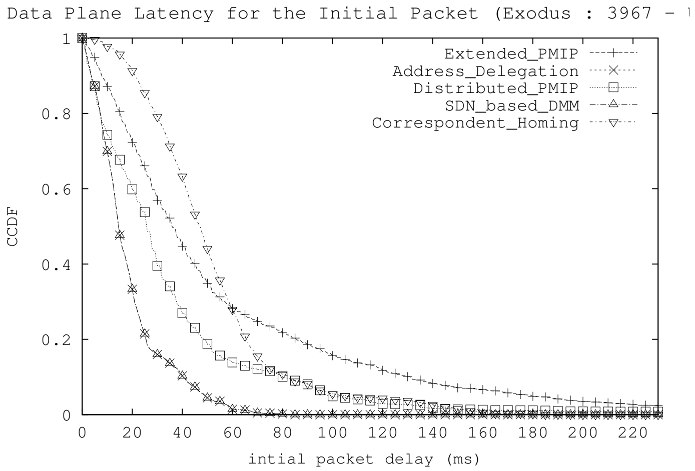


Figure 5.16: Data-plane Latency occurred in Delivering the First Packet : Exodus (3967).

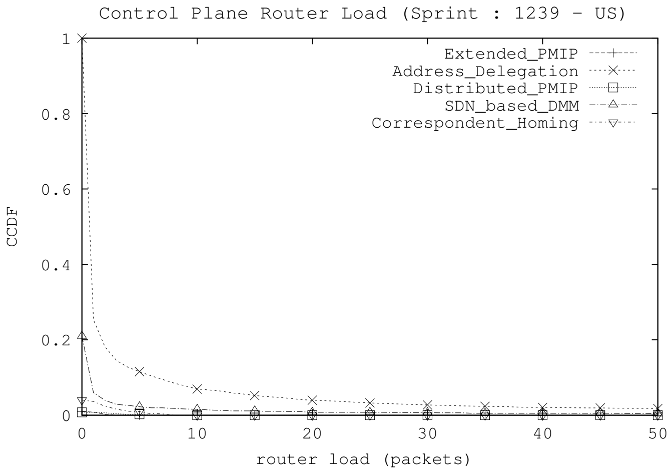


Figure 5.17: Control-plane Router Load : Sprint (1239).

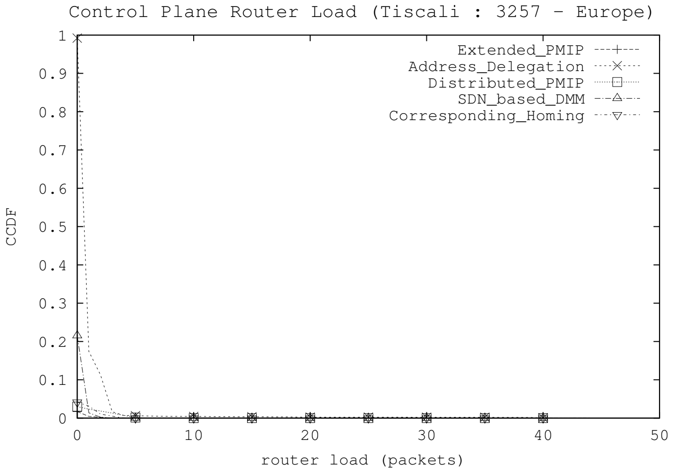


Figure 5.18: Control-plane Router Load : Tiscali (3257).

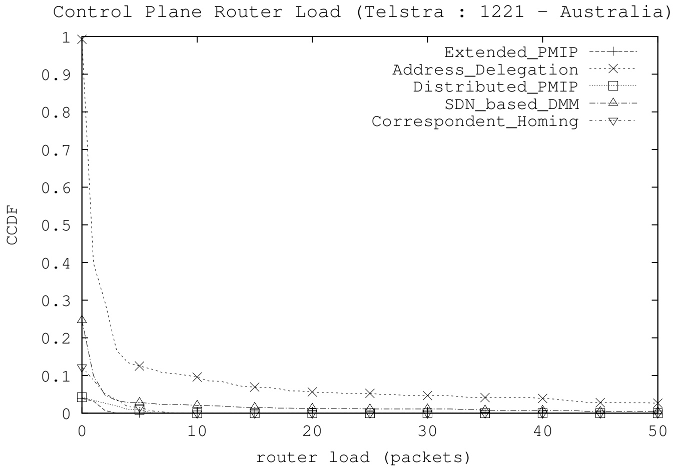


Figure 5.19: Control-plane Router Load : Telstra (1221).

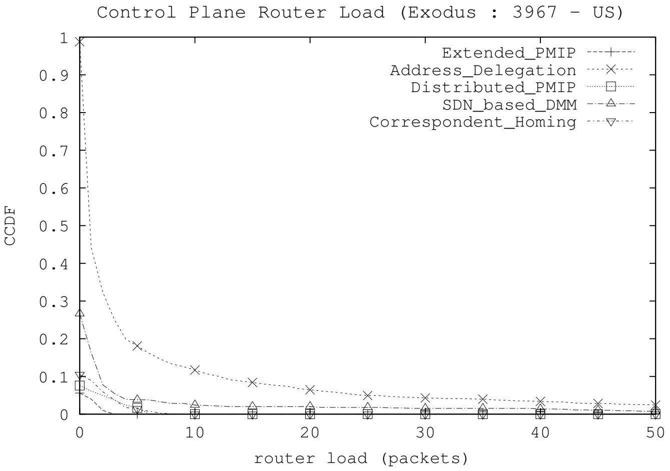


Figure 5.20: Control-plane Router Load : Exodus (3967).

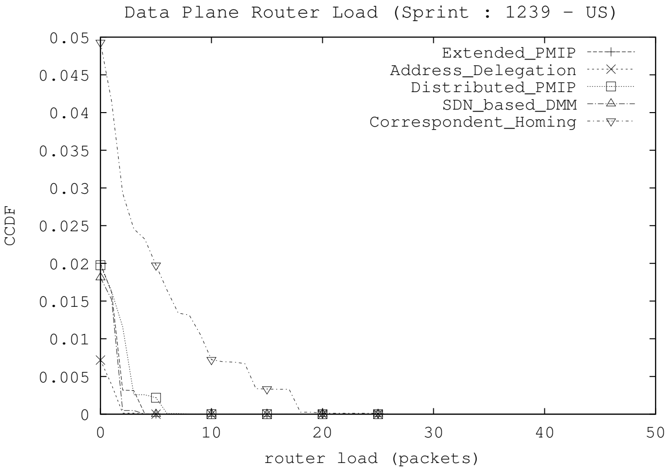


Figure 5.21: Data-plane Router Load : Sprint (1239).

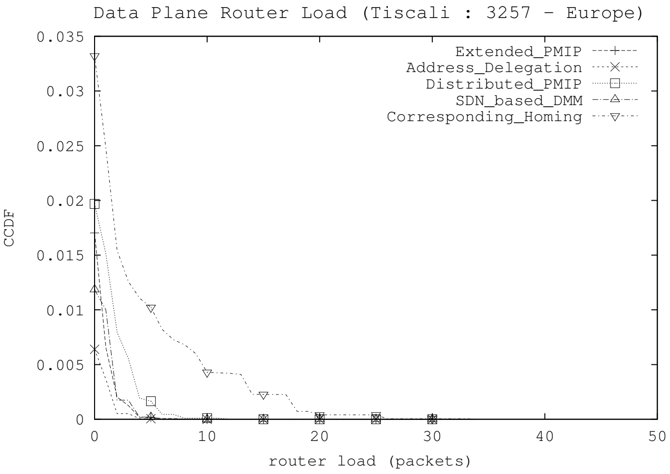


Figure 5.22: Data-plane Router Load : Tiscali (3257).

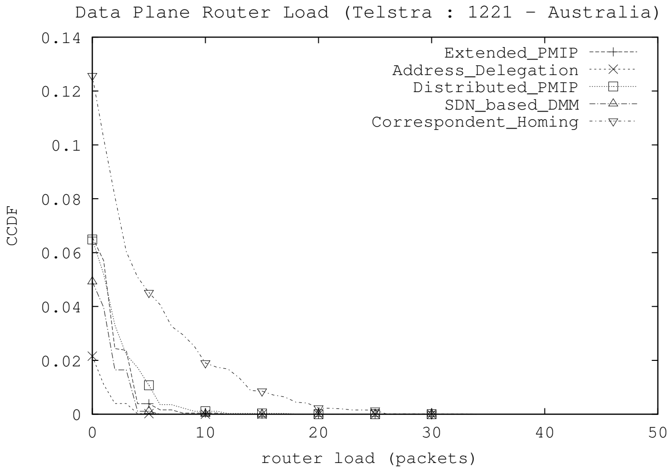


Figure 5.23: Data-plane Router Load : Telstra (1221).

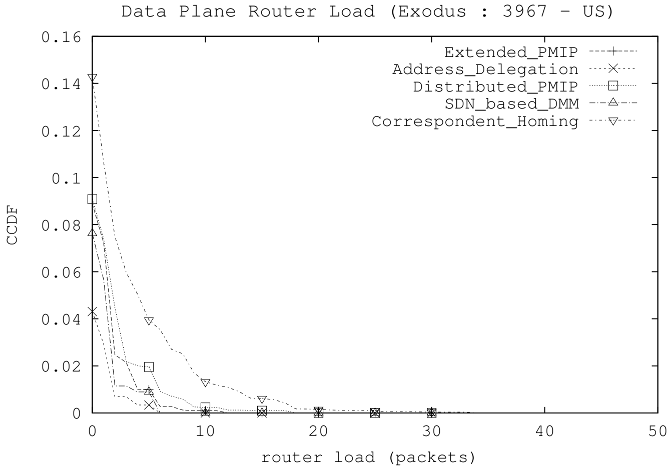


Figure 5.24: Data-plane Router Load : Exodus (3967).

5.4.5 Control-plane Router Load

Figures 5.17-5.20 elaborate the control-plane router loads observed at Sprint, Tiscali, Telstra, and Exodus networks, respectively with the x-axis representing the control-plane router load. Router loads while handover is performed seem to have negligible variances but Correspondent Homing scheme displays a slightly poor router utilization with respect to the other schemes.

5.4.6 Data-plane Router Load

Figures 5.21-5.24 represent the data-plane router loads observed at Sprint, Tiscali, Telstra, and Exodus networks, respectively with x-axis depicting the data-plane router load. Much similar to the control-plane router load, the data-plane router load also shows the least divergence and Correspondent Homing scheme shows comparatively unfavorable results.

Table 5.2: Summary of Results Yielded for the Control-plane.

		Control-plane		
Scheme	AS	Cost	Latency	AS load
Extended PMIP	1239	Best	Best	Better
	3257	Best	Better	Better
	1221	Best	Best	Better
	3967	Best	Better	Better
Address Delegation	1239	Worst	Worse	Moderate
	3257	Worst	Moderate	Moderate
	1221	Worst	Moderate	Moderate
	3967	Worst	Worse	Moderate
Distributed PMIP	1239	Better	Better	Better
	3257	Better	Better	Better
	1221	Moderate	Moderate	Better
	3967	Moderate	Moderate	Better
RO-SDN	1239	Worst	Worse	Better
	3257	Worst	Worst	Better
	1221	Worst	Moderate	Better
	3967	Worst	Best	Better
Correspondent Homing	1239	Moderate	Worse	Better
	3257	Moderate	Better	Better
	1221	Better	Worst	Better
	3967	Better	Moderate	Better

5.5 Overall Findings

5.5.1 Discussion

This section analyses the obtained results based on two perspectives: DMM schemes and ISP topologies. As an insight to the discussion, Tables 5.2 and 5.3 give a summarized view of the results obtained during the evaluation.

5.5.1.1 DMM Schemes

In general, the fully distributed approach which suggests extensions for PMIP, i.e., Extended PMIP scheme and Distributed PMIP scheme, maintain the cheapest control planes with minimal initialization and handover signalling. Both of them use static anchoring closer to the initial location of the MN. This results in longer delays in delivering the data packets and show a slightly unfavorable router load in the data plane as well. Nevertheless, the excessive router load is negligibly small. In contrast, the schemes that consider control / data-plane split, i.e., Address Delegation scheme and RO-SDN, have higher handover costs and latencies while having the least data plane costs and latencies.

Table 5.3: Summary of Results Yielded for the Data-plane.

		Data-plane		
Scheme	AS	Cost	Latency	AS load
Extended PMIP	1239	Moderate	Moderate	Better
	3257	Better	Better	Better
	1221	Worse	Worse	Better
	3967	Worse	Worse	Better
Address Delegation	1239	Best	Best	Best
	3257	Best	Best	Best
	1221	Best	Best	Best
	3967	Best	Best	Best
Distributed PMIP	1239	Better	Better	Better
	3257	Better	Better	Better
	1221	Moderate	Moderate	Better
	3967	Moderate	Moderate	Better
RO-SDN	1239	Best	Best	Best
	3257	Best	Best	Best
	1221	Best	Best	Best
	3967	Best	Best	Best
Correspondent Homing	1239	Worst	Worst	Moderate
	3257	Better	Better	Moderate
	1221	Worse	Worse	Moderate
	3967	Worse	Worse	Moderate

Comparatively, RO-SDN scheme performs slightly better than Address Delegation scheme which makes use of conventional routing table updates as a result of having controller positioned in a much appropriate location chosen based on accessibility. In Address Delegation scheme, it is assumed that any router that facilitates a visiting MN should send out updates to all the other routers. When ARs anchoring the MN reside farther from the core of the network, having least accessibility, they might delay the convergence of routing tables and create an unstable situation where a higher packet loss may occur.

Correspondent Homing scheme fails in showing favorable results during the evaluation although it should theoretically be able to eliminate the problems observed in MIP. This evaluation just assumes higher accessibility in deploying a common CHA leading to a MIP like situation. Specially when the ISPs of higher tiers are concerned, they tend to be distributed in a larger geological area which might even cover different continents. Therefore, employing a single CHA in common can trigger inimical behavior.

The summary performance analysis is given in Tables 5.4 and 5.6. Based on the overall results observed, it can be suggested that schemes which utilize router updates, such as Address Delegation and RO-SDN schemes, seem to fit any topology if the router update mechanism can be regulated to avoid the overheads occur during initialization and handover. Thus, it can be considered a prospective common model to achieve DMM with possible improvements.

5.5.1.2 ISP topologies

According to Tables 5.2 and 5.3, it substantiates the idea that the network topology determines the performance of the deployed scheme to a greater extent. As an ISP is away from the Internet core, it tends to cover small areas with a highly connected topology, which improves the routing performances in general. Thus, the topology has a greater impact on the performance of DMM schemes. Though a certain performance patten is visible in each scheme when they are employed in ISP networks of tier 1 through ISP networks of tier 3, an interruption is observed at the Tiscali network (ASN: 3257). Performance enhancement in each scheme can be seen from Sprint (ASN: 1239) ISP through Exodus (ASN: 3967) in correlation with their size except for the unexpected deviated Tiscali (ASN: 3257). Hence, it suggests that not only the area covered by the ISP, but also the compactness of PoPs has a major impact on mobility protocol performance.

Tiscali network has a minimal number of ARs which provide Internet access to users (Table 3.3) and it is distributed in Europe, which makes it a large scale network residing closer to the Internet core. Thus, it has a poorly connected structure with few high bandwidth links and when the actual geological router topology is inspected, it is seen that the Tiscali network has a limited number of PoPs at each country. This results in having lesser number of ARs within a unit distance. As a result of having a confined set of ARs, the number of candidate sessions is highly limited. Due to the above factors, the Tiscali network itself has perspicuous explanations for the performance observed.

In general, for larger scale ISPs such as Sprint (ASN: 1239) using a centralized anchor as in the Correspondent Homing scheme results in degraded performance. Even Address Delegation and RO-SDN schemes might result in unexpected packet losses due to longer convergence time in such larger networks. In the former case, it suggests that regardless of the scale, for networks that provide both content hosting and access, a generic CHA would not improve local routing. Nevertheless, if an ISP only provides content hosting, or if the CHA placement is regulated according to the location of the content host, then the results might vary considerably. Moreover, while Address Delegation and RO-SDN schemes can be strongly recommended for smaller network away from the core, i.e., for Exodus (ASN: 3967) and Telstra (ASN: 1221), a distributed controller approach may improve the performance of RO-SDN scheme in larger topologies like Sprint (ASN: 1239) or in scattered topologies like Tiscali (ASN: 3257). Among the two schemes, RO-SDN scheme performs better in terms of convergence time which leads to lessen the packet loss that might occur during the handover due to routing table inconsistencies. Alternatively, even with a higher data delivery cost and latency, a scheme like Distributed PMIP scheme may give better reliability in larger scattered topologies but should be determined based on the required quality of service because of their excessive control cost and latency.

The comprehensive summary in Tables 5.2 and 5.3 confirm that factors such as number of routers and links, density of ARs in PoPs, geological scale, and distance from the Internet core of the ISP, largely affect the performance of the employed DMM scheme.

Table 5.4: Summary of the Performance Analysis.

Category	Advantages	Disadvantages
Fully distributed - Extended PMIP scheme - Distributed PMIP scheme	- Least controlling cost/latency	- Higher data delivery cost/latency - Non-optimal data routing - Higher router load
Control/data-plane split - Address Delegation scheme - RO-SDN scheme	- Least data delivery cost/latency - Optimal data routing	- Higher controlling cost/latency - Possible packet loss due to longer convergence time
Destination-end anchoring - Correspondent Homing	- Static anchor point	- Higher controlling cost/latency - Higher data delivery cost/latency - Non-optimal data routing - Possible single point of failure

In general, schemes based on router updates perform better with any ISP in overall. But, as already mentioned, they should be enhanced to accommodate larger ISPs in an economical manner. Thus, this evaluation suggests that improvement of convergence time by regulating the router update mechanism and by bringing the updating entity closer to the subjected routers might be potential considerations where further research should be concentrated.

5.6 Summary

This chapter analyzed the performance of five DMM schemes, which inherit three major properties. One category inherits fully distributed approach, where both the control and data-planes are distributed. Those are, Extended PMIP and Distributed PMIP schemes. The second category inherits control and data-plane split. RO-SDN and Address Delegation schemes fall into this category. Finally, the Correspondent Homing inherits destination-end anchoring.

Results of the evaluation suggest that behavior of DMM schemes differ based on ISP properties. These properties include number of routers and links, density of ARs in PoPs, geological scale, and distance from the Internet core. Although PMIPv6 based schemes can be recommended for smaller topologies like Telstra (ASN: 1221) and Exodus (ASN: 3967), they suffer different consequences like non optimal routing and scalability due to anchoring, thus limiting the performance in larger and scattered topologies such as Sprint (ASN : 1239) and Tiscali (ASN: 3257). Address Delegation and RO-SDN schemes outcast their impoverished control-plane performance with attaining optimal routing at the date-plane, hence being successive in topologies of any scale. However, the latency in routing table convergence might result in packet losses in larger networks if not addressed with appropriate precautions. Alternatively, the PMIPv6 based schemes can be employed in larger networks to achieve higher reliability but it results in longer data delivery time, thus the selection of the scheme should be done according to the prescribed quality of service of the ISP. Correspondent Homing scheme displays degraded performance than the theoretical expectation as a result of centralized anchoring assumed during the evaluation. Thus, it does not suit any topology with a common anchor. Looking at the comprehensive outcomes, it can be suggested that decoupling

Table 5.5: Summary of the Performance Analysis.

Category	Suitable Topologies	Unsuitable Topologies
Fully distributed - Extended PMIP scheme - Distributed PMIP scheme	- Smaller topologies with least traffic	- Scattered topologies - Topologies with higher traffic
Control/data-plane split - Address Delegation scheme - RO-SDN scheme	- Smaller to medium topologies - Topologies with higher traffic - Larger scattered topologies with distributed controlling	- Larger scattered topologies with a central controller
Destination-end anchoring - Correspondent Homing	- Topologies allowing wide scope of mobility - Scattered topologies with distributed anchoring	- Larger Scattered topologies with a central anchor

the control and data-planes is an effective approach for DMM. Nevertheless, due to the enormous delays and costs encountered in performing the handover, a possible setback that should be addressed in order to make such DMM schemes perform better in both the planes is highlighted. Regulation of route updates and distribution of entity which sends off the updates were suggested as feasible enhancements.

Chapter 6

Design and Evaluation of DMMSDN

6.1 Motivation

Based on the outcomes of the evaluations described in Chapts. 4 and 5, it can be identified that fully distributed MM and control/data-plane split achieve the best control and data plane performances. However, the proposals found in the literature lack composite optimization of both planes. Table 6.1 shows the overall findings of the evaluations described earlier in Chapts. 4 and 5. According to Table 6.1, the schemes which attain the best results in the data plane tend to have heavy overhead in the control-plane. On the other hand, the schemes with best control-plane performance tend to perform unsatisfactorily in the data-plane. It questions the feasibility of using the proposed schemes, when the utilization of both the planes are considered. Thus, a sustainable DMM scheme which optimally exploits both the planes is highly desirable.

6.2 Requirements

Understanding the requirement of DMM is extremely important in designing a sustainable solution. The functional requirements that are introduced earlier in Sect. 1.2.1 are valid for any MM scheme. Considering the provision of mobility transparent to other layers, the candidate scheme should consider the methodology of providing an unchanged IPv4 / IPv6 address to the MN. When achieving that, the scheme should make sure it does not introduce an overhead or modifications for stationary nodes. Further, when the MN does not require mobility or when it is at the home network, the DMM scheme should make sure that it does not introduce unnecessary DMM overhead.

Table 6.1: Overall Best Results of the conducted Evaluations.

		Global DMM Schemes	Localize DMM Schemes
Control-plane	Best	Functional distribution	Fully distribution
	Worst	Fully distribution	Control/Data-plane split
Data-plane	Best	Return Routability	Control/Data-plane split
	Worst	Centralized MM	Fully distribution

Reducing non-optimal routes can also be considered as one of the main objectives of a new scheme. Most importantly, scalability should also be given high attention, due to the dramatic demand for mobility.

Apart from the general MM requirements, the DMM-WG [21] introduces the following set of requirements that should be fulfilled by DMM in their recent RFC document [59].

- Distributed mobility management
- Bypassable network-layer mobility support for each application session
- Deployment in IPv6 environment not in IPv4 environment
- Existing mobility protocols
- Coexistence with deployed networks/hosts and operability across different networks
- Operation and management considerations
- Security considerations
- Multicast considerations

In addition to the general functional considerations introduced above, it is required to identify the design requirements of DMM. Unlike the standard protocols where there is a single agent to maintain MM information, distribution of MMEs introduces different design ramifications. For instance, it is required to thoroughly conceive a mechanism to achieve the distribution. Distribution of MMEs can be done in two different ways.

1. Duplication :
   MME is duplicated and distributed in the domain. Each MME is capable of performing similar tasks. Data redundancy is one of the main features of this approach. Thus, it reduces the vulnerability to encounter a failure. However, it requires a complex mechanism to maintain the consistency of mobility information. As a result, it might introduce a high controlling overhead.

2. Functional Distribution :
   In this approach, different functionalities of the MM are identified and distributed. For instance, location data of the MN can be handled by a single entity, while other functions such as mobility routing and address allocation can be distributed. The functionality of each entity is specific in this architecture. Thus, the network resource allocation can be optimized by allocating specific resources to distributed MMEs. However, it requires collaboration of each of these entities in order to handle mobility. This might also introduce a controlling overhead.

On one hand, distribution reduces the risk of a single point of failure and attack. At the same time, it requires multiple establishments where the initial cost might be higher. Thus, having an excessive number of MMEs might introduce unnecessary cost. On the other hand, achieving DMM with duplication or functional distribution introduces controlling overhead. Thus, the design should be enhanced with minimal controlling overhead. Followings can be considered as major design requirements.

- Determining target mobility scope and the network :
  In order to define a sustainable approach to handle mobility, it remains compulsory.

- Optimal functional distribution :
  In order to determine the economical functional distribution, it is necessary to determine the functional granularity of MM. The amount of functions that are placed on each MME would contribute to determine the controlling overhead of MM. Thus, the determination of optimal functional coherence and co-location is vital.

- Consistency handling :
  In order to maintain the quality of service, the accuracy of MM information is crucial. Therefore, there should be a firm process to handle consistency of MM information.

- Scalability :
  Increasing demand for the mobility urges a scalable MM architecture. The design should complement transparent and smooth scalability.

- Route optimization in data-plane.

- Least control-plane overhead :
  It is an obvious fact that DMM introduces controlling overhead. Inter-MME communication may occupy a large portion of DMM signalling. The design should not let this overhead to exceed that of the current practices.

- Optimal degree of MME distribution
  Having excessive MMEs would degrade the network performance due to control overhead. Thus, it is vital to determine the optimal number of MMEs that should be installed.

6.3 Overview of DMMSDN

This sections gives a deign overview of the proposed DMMSDN scheme. The functional specification of DMMSDN is provided in Sect. 6.5.

Disadvantages of the current proposals encountered during the evaluation in Chapts. 4 and 5, urge the requirement to consider a sustainable methodology to achieve DMM. The overall finding suggests that a fully distributed scheme which is rooted in the control/data-plane split might possibly address the DMM considerations better. Further, considering the requirements introduced in Sect. 6.2, the basic outline of the novel proposal is defined as follows.

Considering the requirements already described in the Sect. 6.2, DMMSDN looks at the candidate network architecture and desired scope of mobility that is required to be addressed. A comparison of the global and localized MM schemes are given in Table 6.2. Looking at the factors like security threats, controlling overhead, and required modifications, network-based approach is seen enviable. In the current context, users tend to

Table 6.2: Comparison of Global and Localized Mobility Management Schemes.

Feature	**Global** MM	**Localized MM**
Scope of mobility	No restriction	within local domain
Mobility transparency	No	Yes
Mobility related signalling	Between MN and network entity	Only within network
Tunneling	MN to network entity	Within network
Handover latency	increased	Improved
Location privacy	Not provided	Provided
Lagecy devices	Not supported	Supported

move within the same ISP domain, rather than moving across domains. This also complements the network-based approach. Since the inter-domain mobility handling cannot be easily supported by network-based schemes. In order to facilitate network-based inter-domain MM, it will require the negotiation between different domains. However, if the MM schemes can easily be extended to facilitate global mobility, it can be considered as an advantage. Thus, DMMSDN focuses on providing scope extendability as well.

Software Defined Networking (SDN) [1] is considered as the underlying technology for DMMSDN. SDN provides the capability of separating control and data-planes. An overview of the SDN paradigm is given in Sect. 6.4.1. In current mobile networks, control and data-plane separation is successfully obtained. Thus, the applicability of control and data-plane split can be justified. Reason for selecting SDN can be emphasized as the simplicity it provides for the separation. SDN logically centralizes the control-plane of the network in the SDN-controller whereas the data-plane, known as the forwarding plane, simply implements the decisions made by the centralized SDN-controller. Moreover, it simplifies the employment of network functionalities such as mobility. Network functions are only required to be programmed in the SDN-controller. No modifications are required in the other network devices.

After determining the target network and the scope of the mobility, it is required to determine the general functions of MM in order to resolve the optimized distribution of functions. There are several functions that can be identified that present in any network-based MM scheme.

- MN authentication
- Assigning mobility enabled IPv6 addresses to MNs
- Keeping tracks of the MN
- Reporting states of MN to the mobility tracker
- Mobility routing

The followings describe the design approaches followed in addressing the above functional blocks.

6.3.1 MN authentication

On top of the underlying SDN paradigm, Dynamic Host Configuration Protocol for IPv6 (DHCPv6) is utilized in DMMSDN to assign IPv6 addresses to MNs. Apart from address allocation, DHCPv6 is entrusted with the task of accounting and security. At this level, L2 address is used to identify and authenticate MNs. No modifications are introduced to DHCPv6. All the switches which provide access to MNs (ARs) are assumed to send Router Advertisement (RA) messages which have the *M flag* set to 1. This RA forces the MN to adopt DHCPv6 in acquiring an IPv6 address.

6.3.2 Assigning Mobility Enabled IPv6 Address to the MN

In the general SDN architecture, a logically centralized SDN-controller is assumed to perform controlling tasks on behalf of all the entities in the network. However, depending on the size of the network, an ideally centralized SDN-controller might take the form of physically distributed, logically centralized SDN-controller in the actual implementation environment. The idea of SDN requires all these SDN-controllers to maintain a general blueprint of the whole network and to know all the events take place in the network. As long as the mobility is concerned, each node which demands for mobility should be facilitated with the MM by keeping up-to-date tracks of the MNs. This occupies a vast memory and functional portion of the SDN-controller. Further, it might degrade the overall performance by requiring frequent message exchanges between SDN-controllers to update states related to MM.

DMMSDN proposes blocking the domain and assigning a single physical SDN-controller to handle mobility on behalf of the MNs in a well-defined single block. This block is termed as the *area network* in DMMSDN. This is similar to the sub-networking in the IP networking. IPv6 address assigned to a node is used to identify the area network to which the MN belongs. Further, it can be used to resolve the nodes which require mobility, nodes which are mobile but those do not require MM, and stationary nodes. That is by keeping separate IPv6 address blocks to address different types of nodes in the network. Thereby, unnecessary overhead that might occur due to introducing MM can be avoided. Figure 6.1 depicts an example of address categorization mechanism adopted in DMMSDN.

DMMSDN prefers the SDN-Switches in the SDN environment to advertise the same network prefix and the presence of DHCPv6 servers for address configuration (Sect. 6.3.1). Thus, the MN does not detect its movement. Initially, the MN requests IPv6 address configuration. All the SDN-controllers can act as DHCPv6 servers independently. Since the IPv6 address range that can be offered by each SDN-controller is firmly defined. However, DMMSDN requires a mechanism to make sure the specific MN which requests the IPv6 address configuration is not assigned an IPv6 address before. Thus, a table is introduced where the L2 address of the MN is stored along with its already assigned IPv6 address. This table is replicated in all the SDN-controllers. Thus, whenever a certain MN is assigned an IPv6 address, the SDN-controller that assigns the IPv6 address updates all the other SDN-controllers about L2-IPv6 address binding of the MN.

DMMSDN suggests storing mobility information of an MN only in one SDN-controller. Instead of redundantly storing them in all the SDN-controllers. This is supposed to en-

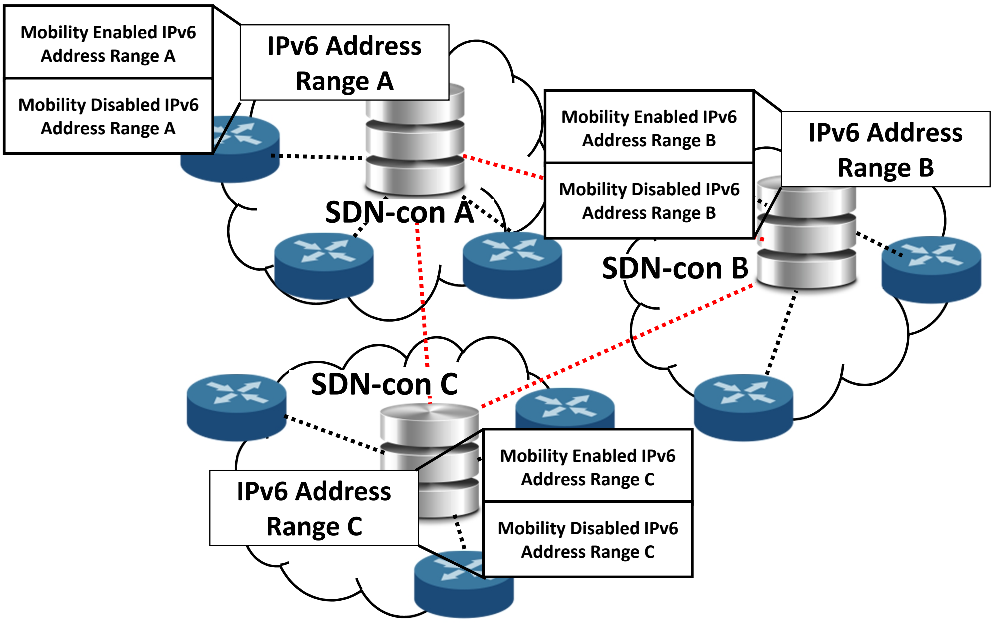

Mobility Enabled IPv6 Address Range A
Mobility Disabled IPv6 Address Range A
IPv6 Address Range A
SDN-con A
Mobility Enabled IPv6 Address Range B
Mobility Disabled IPv6 Address Range B
IPv6 Address Range B
SDN-con B
SDN-con C
IPv6 Address Range C
Mobility Enabled IPv6 Address Range C
Mobility Disabled IPv6 Address Range C

Figure 6.1: Address Categorization Mechanism.

sure the consistency of the mobility information. On the other hand, since it does not require the SDN-controllers to exchange mobility information frequently to ensure the consistency, it is expected to reduce the controlling overhead. Further, the memory occupied by DMMSDN is anticipated to reduce dramatically, as the information volume will inversely correlated with the number of the SDN-controllers in the domain. It is as a result that the MNs in the domain are divided between the SDN-controllers.

6.3.3 Reporting States of MN to the MME

In order to provide mobility to MNs, it is vital to define a mechanism to detect and report MN's events to the MME. For example, if the MN detaches from the network, and re-attaches, it should be detected efficiently. In host-based MM, the MN identifies its mobility and reports to the agent. However, in network-based MM, network should employ special entities which can detect it. Thus, in general schemes, it is required to modify all the ARs which might provide access for MNs. SDN has special features which simplifies this situation. As a result of detaching the decision making from the switch, the only requirement to detect such events is to make AR forward the control over to the SDN-controller at the attachment. Therefore, DMMSDN suggests using DHCPv6 procedure to detect the initial attachment and reattachment.

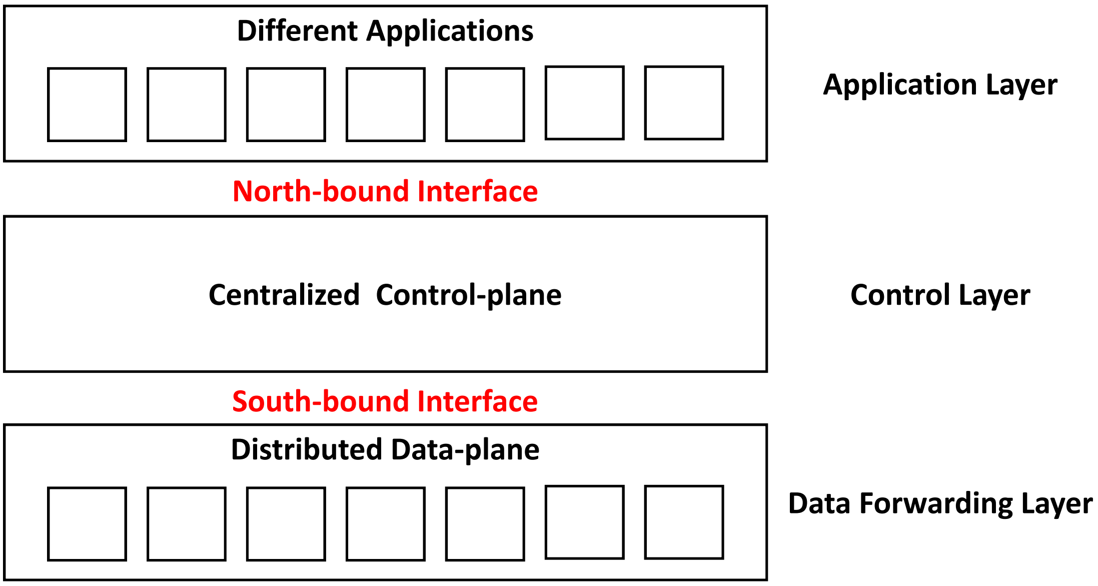


Figure 6.2: Reference Model of SDN.

6.3.4 Mobility routing

SDN streamlines MM with its dynamic, programmable flow alteration mechanism. This allows the SDN-controller to determine and to set routes between the nodes dynamically, when a movement is detected. In order to allow the mobility handling to function without a fail, DHCPv6 and the SDN-Switch functionalities are required. The detection of MN's events like attachment and detachment is supported by DHCPv6 where the SDN-controller is assumed to act as the DHCPv6 server. However, no modifications are introduced to DHCPv6. Requirement to DMMSDN to perform the mobility handling is restricted to the ability of the AR to forward the necessary packets to the SDN-controller. Thus, no modifications are introduced to the SDN paradigm.

6.4 Mobility Management in SDN Environment

In order to understand the specification of DMMSDN, first, it is required to have a general idea about SDN paradigm. Thus, this section introduces SDN briefly.

6.4.1 Functional Overview of SDN

Software Defined Networking (SDN) [1] can be considered as one of the most trendy paradigms to separate control and data-planes. Figure 6.2 shows the reference model of SDN.

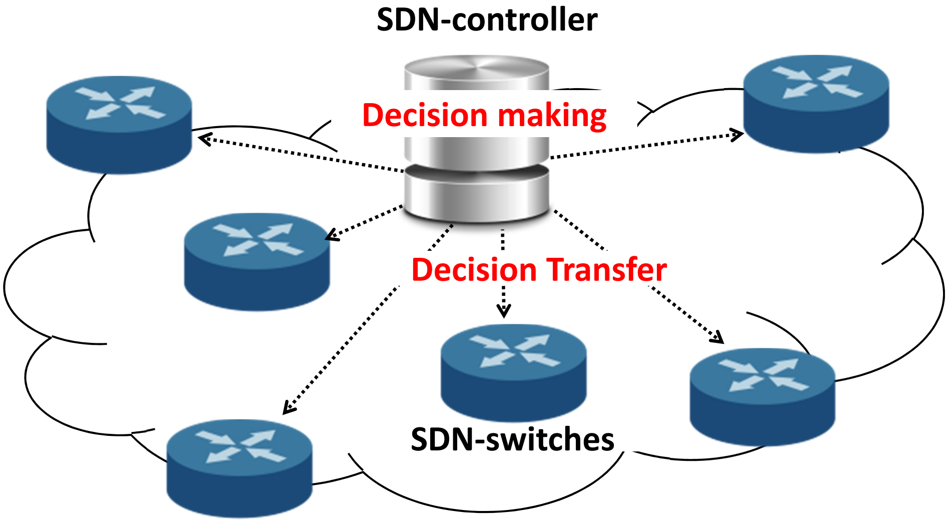


Figure 6.3: General Functionality of SDN.

It suggests using a logically centralized control-plane. By utilizing a logically centralized SDN-controller (Fig. 6.3 : (1)), it suggests that the networking control decisions can be taken efficiently. Unlike the routers that present in current networks, the controlling functionality is separated and pushed to the SDN-controller. Thus, the routers in the SDN environment are simple switches, which act according to the decisions fed by the SDN-controller. The SDN-controller is responsible for determining all the flows in the network. As the controller knows all the information about the flows in the network, it has the ability to utilize the resources in the network in an optimized manner. This can be considered as the main advantage of the concept of SDN. It allows the SDN-controller to provide resources on-demand while perform automated load balancing. Further, it enables to scale network resources easily.

On the other hand, by concentrating the network controlling at the logically centralized SDN-controller, it allows the easy deployment of network functionalities. In order to introduce new functioanilities into the network, it does not require the modification of all the routers. Instead, it is sufficient enough to introduce the functionality to the SDN-controller. Thus, this simplifies the network modifications. On-demand resource provisioning, automated load balancing, flexibility of network administration, and the ability to scale the network resources with application and data needs can be considered as few the advantages of SDN.

6.4.2 Distributed Mobility Management in the SDN

Figure 6.4 shows the basic SDN architecture that enables MM. As already mentioned, employment of MM in the SDN environment is much simplified due to the ability to employ it without modifying the network infrastructure. The MM functionality is introduced only to the SDN-controller. Further, Fig. 6.5, shows an instance where the

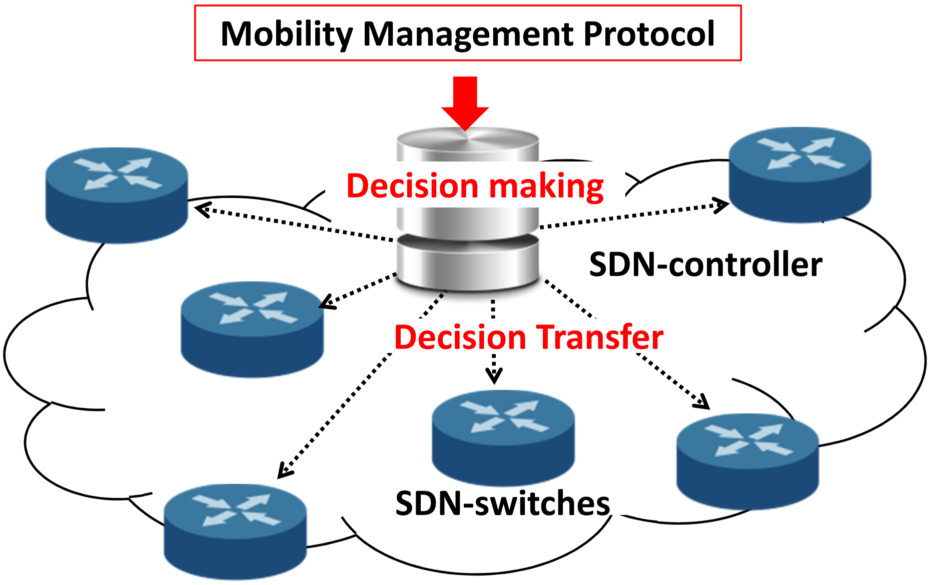


Figure 6.4: General MM Employment in SDN.

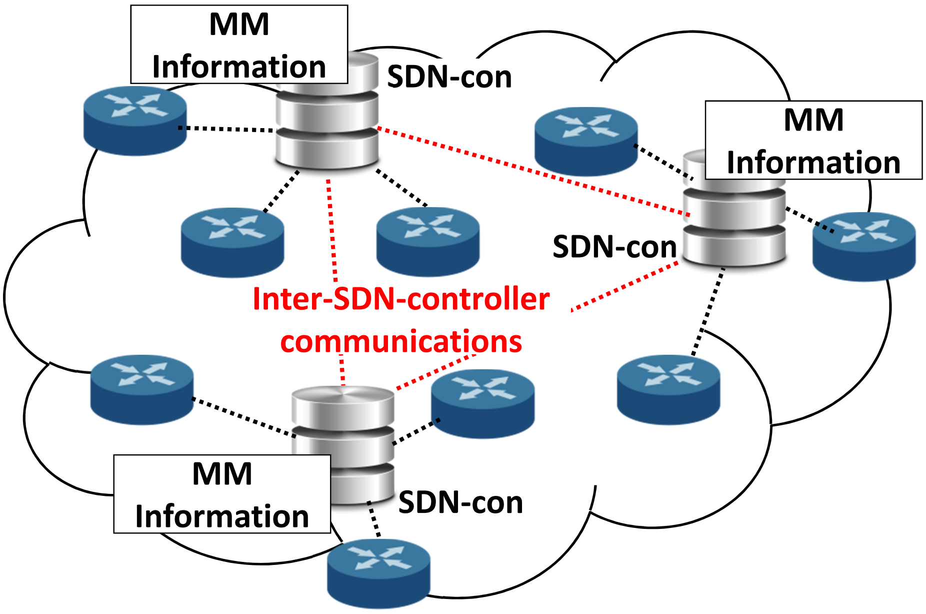


Figure 6.5: DMM with SDN.

DMM is employed in the SDN environment. Even though the SDN paradigm suggests a logically centralized SDN-controller, it might take the form of physically distributed collaborative SDN-controllers, depending on the implementation. For instance, if the size of the network is large, it might be difficult for a single SDN-controller to control the whole network. Therefore, the implementation can be based on a couple of physical SDN-controllers, which function as a single logical controller. This will ultimately imply that in order to facilitate the MNs, all the mobility information should reside in all the SDN-controllers. Thus, data redundancy occurs. Further, it results in an overhead in the control-plane. This is due to the intercommunication between the SDN-controllers to maintain consistency of the network information.

6.5 Specification of DMMSDN

6.5.1 Terminology

Apart from the general terminology used in the MM, DMMSDN introduces some new terms used to identify entities and concepts. Figure 6.6 shows the general scenario which is considered for the terminology.

The first AR to which the MN attaches is identified as the *Initial AR (I-AR)* and the SDN-controller which manages the I-AR is known as the *Initial Controller (I-Con)*. Similarly, the AR and the SDN-controller serving the CN are known as the *CN-AR* and the *CN-Con*, respectively. A subsequent attachment of the MN to the ISP network via a different AR results in a new serving AR identified as *Serving AR (S-AR)*. The SDN-controller facilitating the S-AR is known as the *serving SDN-controller (S-Con)*. The terms *Any-Con* and *Any-AR* are used to identify any other SDN-controller or AR except for the ones defined above.

6.5.2 Memory Requirements of DMMSDN

In order to maintain mobility information, DMMSDN proposes the installment of several memory entities as given in Fig. 6.7. Following gives a brief overview of those memory entities.

1. *L2-IPv6 Binding List (L2-IPv6 List)* :
   Resides in all SDN-controllers. The L2-IPv6 List stores hard-state *L2-IPv6 Binding List Entries (L2-IPv6 Entries)* that act as a repository to identify IPv6 addresses that are already assigned to MNs.

2. *Binding Cache List (BCL)* :
   Resides in I-Con, comprises soft-state *Binding Cache Entries (BCEs)* each of which represents the *binding* of the MN's initial address and its current AR.

3. *Binding Request List (BRL)* :
   Resides in I-Con, contains soft-state *Binding Request Entries (BREs)* which represent the *requests* arrived inquiring the MN's binding.

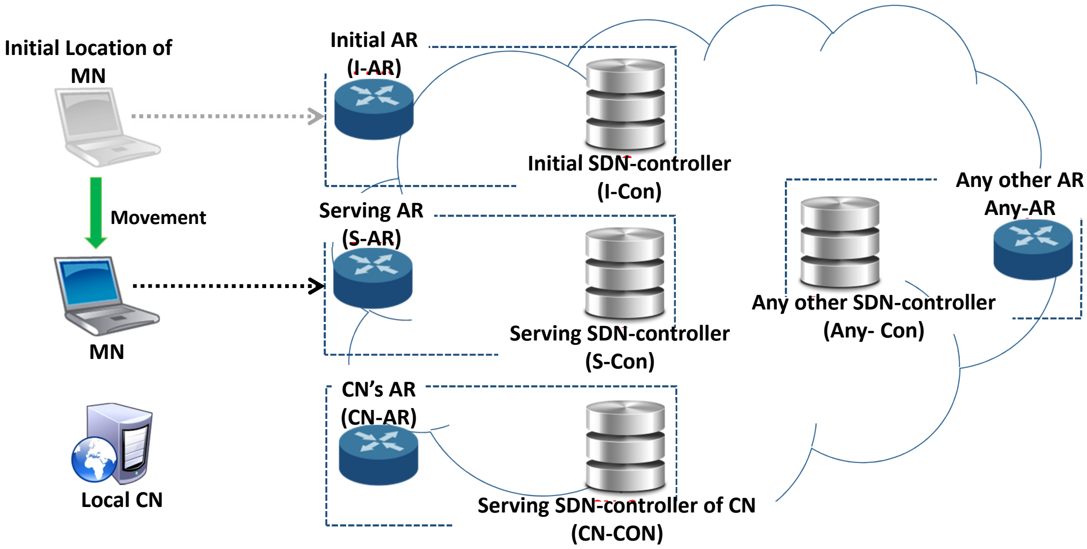
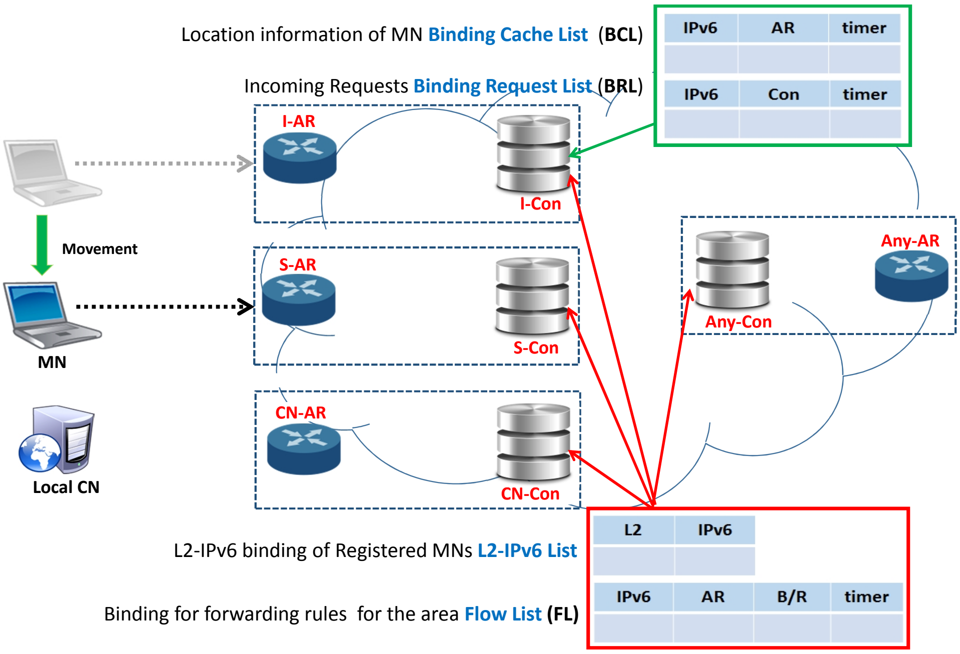


Figure 6.6: Terminology.


Location information of MN Binding Cache List (BCL)
Incoming Requests Binding Request List (BRL)
I-AR
I-Con
Movement
S-AR
S-Con
Any-AR
Any-Con
MN
CN-AR
CN-Con
Local CN
L2-IPv6 binding of Registered MNs L2-IPv6 List
Binding for forwarding rules for the area Flow List (FL)

IPv6	AR	timer

IPv6	Con	timer

L2	IPv6

IPv6	AR	B/R	timer

Figure 6.7: Memory requirements of DMMSDN.

Table 6.3: Space Required for Each Entry.

SDN-Con.	Memory	Entry	Field	Type	Size (bytes)
I-Con	BCL	BCE	$IPv6_{MN}$	IPv6 addr.	16
			$IPv6_{AR}$	IPv6 addr.	16
			Timer	seconds	2
	BRL	BRE	$IPv6_{any}$	IPv6 addr.	16
			Con	ID	1
			Timer	seconds	2
All-Con	L2-IPv6 List	L2-IPv6 Entry	L2	L2 addr.	6
			IPv6	IPv6 addr.	16
	FL	FLE	$IPv6_{any}$	IPv6 addr.	16
			$IPv6_{AR}$	IPv6 addr.	16
			B/R	boolean	1
			Timer	seconds	2

4. *Flow List (FL)* :
   Accommodates soft-state *Flow List Entries (FLEs)* for each inquiry performed, and bindings maintained by each SDN-controller.

Table 6.3 depicts the space occupied in all the SDN-controllers by each entry described above, which is illustrated in Fig. 6.7.

6.5.3 Mobility Management with DMMSDN

The functionality of the DMMSDN can be identified around three main events.

1. Initialization
   The MN attaches to the SDN network and performs data communication with the CN

2. Handover
   The MN detaches from the initial location and relocates in a different location. Session continuity of the sessions that are started at the previous location is provided at the new location.

3. De-registration
   The MN deregisters from the mobility handling procedure provided. It results in discontinuity of all the established sessions, which were supported by DMMSDN.

The specification of each of these functions are explained in Sects. 6.5.3.1 to 6.5.3.3.

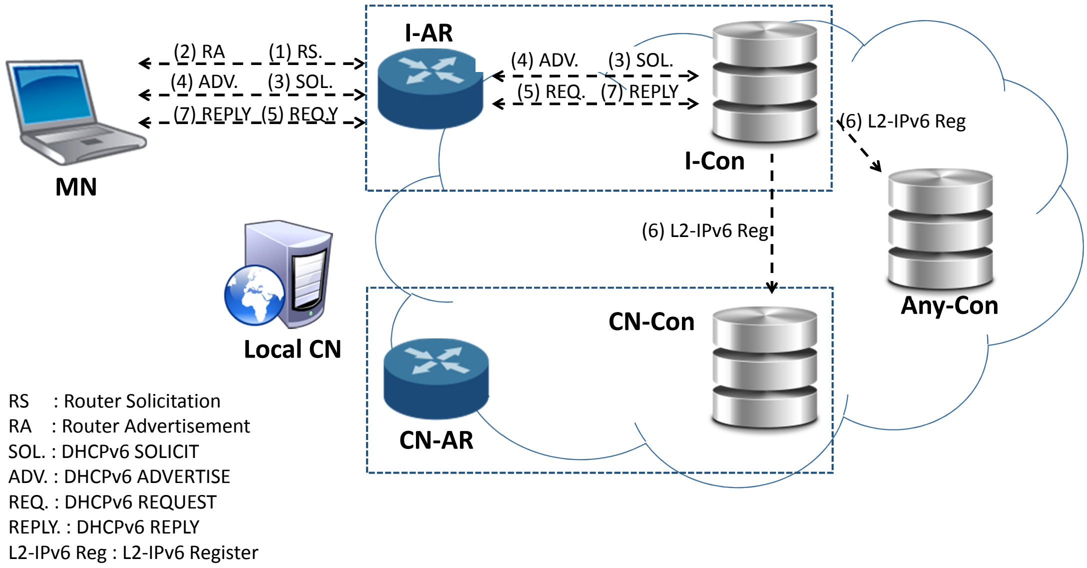


Figure 6.8: Initialization.

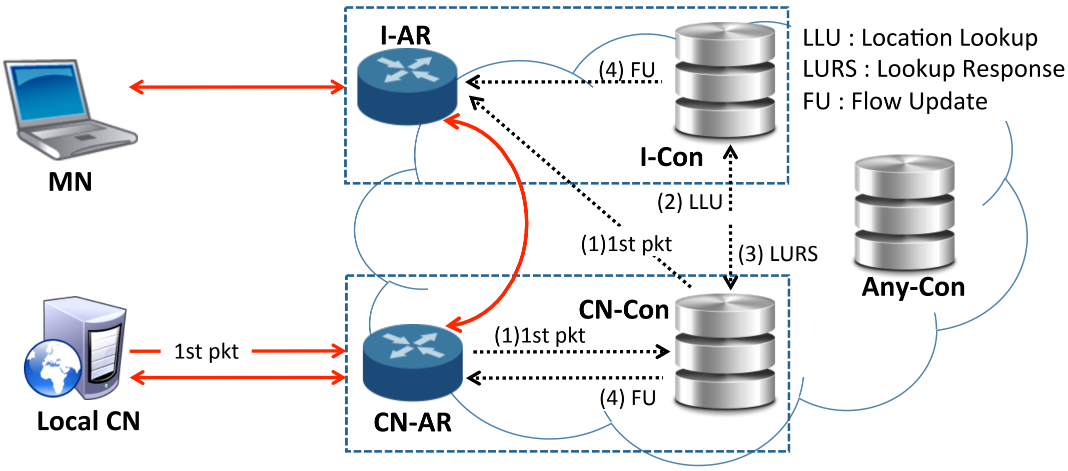


Figure 6.9: Data Communication Between MN and CN at the Initial location.

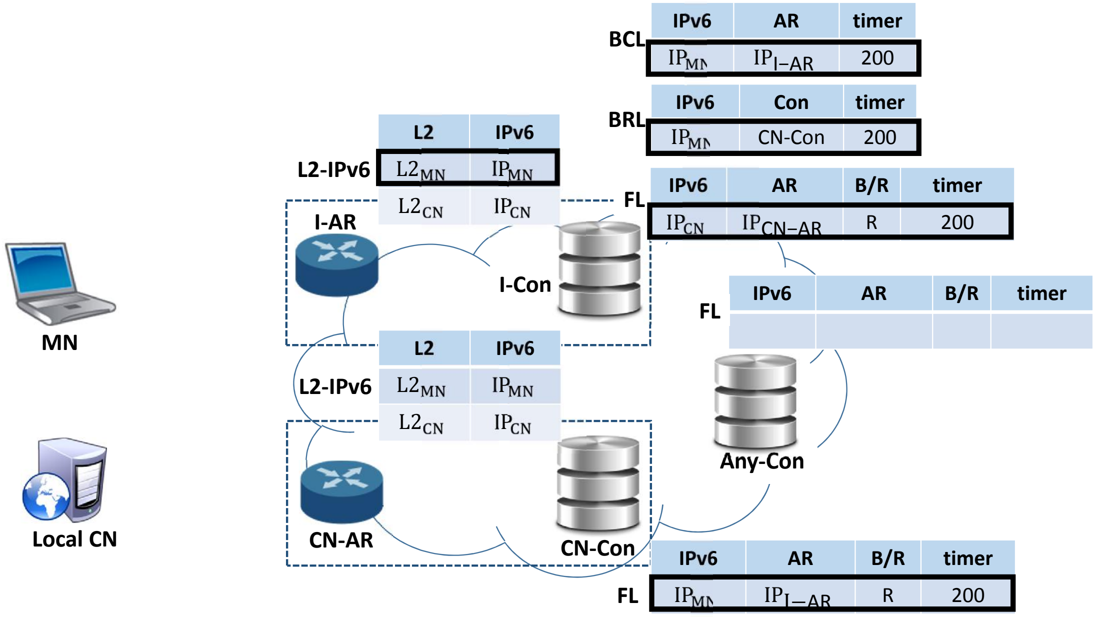


Figure 6.10: Memory Involved During Initialization.

6.5.3.1 Initialization and data communication

The control and data flow during initialization and data communication are shown in Figs.6.8 and 6.9, respectively. The memory involved during the initialization and the data communication at the initial location are depicted in Fig. 6.10.

Initially, the MN attaches to an AR, which is identified as the I-AR, and exchanges Router Solicitation (RS) and Router Advertisement (RA) messages (Fig.6.8 (1) and (2)). The RA message has the *M flag* set to 1, by which the I-AR forces the MN to adopt DHCPv6 in acquiring an IPv6 address. DMMSDN further suggests that the all RA messages should contain the same L2 address in the source link-layer option. At the same time, it suggests that the same network-prefix should be included in the prefix information option.

In response, the MN sends the *SOLICIT* message (Fig.6.8(3)) in order to discover DHCPv6 servers. This message is intercepted by the AR (I-AR). Since the I-AR is not configured to handle such messages, it is relayed to the I-Con. At the reception of the *SOLICIT* message, the I-Con first checks its L2-IPv6 List for any IPv6 address previously assigned to the MN, using MN's L2 address as the key. If no entry is found, then the I-Con sends the *ADVERTISE* message (Fig.6.8(4)) offering an IPv6 address and configuring parameters to the MN via the I-AR.

The MN responds with the *REQUEST* message (Fig.6.8(5)), which is relayed to the I-Con through the I-AR. Then, the IPv6 address is granted to the MN (Fig.6.10(2) in the I-Con). At the same time, the binding of the L2 and IPv6 addresses of MN is registered in the *L2-IPv6 List* as shown in Fig.6.10(2).

Subsequently, I-Con creates a BCE (Fig.6.10(1)) on behalf of the MN. BCE is supposed to keep the records of the AR to which the MN is attached to. Then, using the *L2-IPv6 Reg* message (Fig.6.8(6)), the I-Con informs the other SDN-controllers of the registration (Fig.6.10(3)). Thus, all the SDN-controllers will register the *L2-IPv6* binding of the MN. At the same time, the *REPLY* message (Fig.6.8(7)) is also sent to the MN by the I-Con which is relayed via the I-AR. This completes the initialization and the IPv6 address that is assigned to the MN is mobility enabled.

When the CN attempts to send data-packets to the MN, on the arrival of the first packet at the CN-AR, the CN-AR forwards it to the CN-Con due to the absence of a correspondent flow table entry. The CN-Con resolves the SDN-controller which maintains BCE of the MN by inspecting the destination IPv6 address. This is possible since it is aware of the *IPv6 address ranges* managed by all the SDN-controllers in the ISP network (Fig.6.10(3)). Then, the CN-Con sends the *Location Lookup* message to the I-Con, inquiring the current location of the MN. At the same time, CN-Con includes an FLE in its FL as depicted in Fig.6.10(4). The I-Con responds with the *Lookup Response* message, which indicates where the MN resides and at the same time it inserts the information of CN-Con n its BRL, as a BRE of the MN (Fig.6.10(5)).

Finally, the I-Con triggers the flow table updates for the routers lying on the path between MN and CN. Flow table updates of a certain SDN-switch is always triggered by the SDN-controller of the *area network* to which the SDN-switch belongs to. As the overall topology of the local network is known by all the SDN-controllers, the I-Con is capable of determining the path between the given CN and the MN. However, the exact path is not necessarily pre-determined. The network administrator can simplify the process by allowing the SDN-controller to determine the *area networks* that would be possibly traversed by a packet exchanged between CN and MN. This may heavily dependent on routing policies adopted by the local network. Then, the I-Con informs the SDN-controllers of those *area networks* (Any-Con) about the prospective communication between CN and MN using the *Flow Update Request* message. This is the proactive flow table update method. After receiving the *Flow Update Request* message, the Any-Con determines the required SDN-switches to be updated considering the policies and topology. Then, *Flow Update* takes place. For each of the Any-Con that is being informed about the flow between CN and MN, the I-Con updates the BRL entry of the MN. This update contains the SDN-controller information of which are affected, that required to be updated. As a result, those Any-Cons also include soft-state entries in the FL for the MN as *request*. Even if the I-Con fails to determine other SDN-controllers that should be updated, the flow-updates will take place in a reactive manner. That is, when a packet arrives at a certain SDN-switch, it will forward it to the SDN-controller of its *area network*, then the SDN-controller will inquire the binding information of the destination in the similar way.

On the reception of the *Lookup Response* message, the CN-Con forwards the first packet(s) already forwarded by the CN-AR to the learnt location of the MN. Subsequent packets will follow the optimal path between the CN-AR and the I-AR, that is effectively set by the flow table updates which is previously triggered by the I-Con.

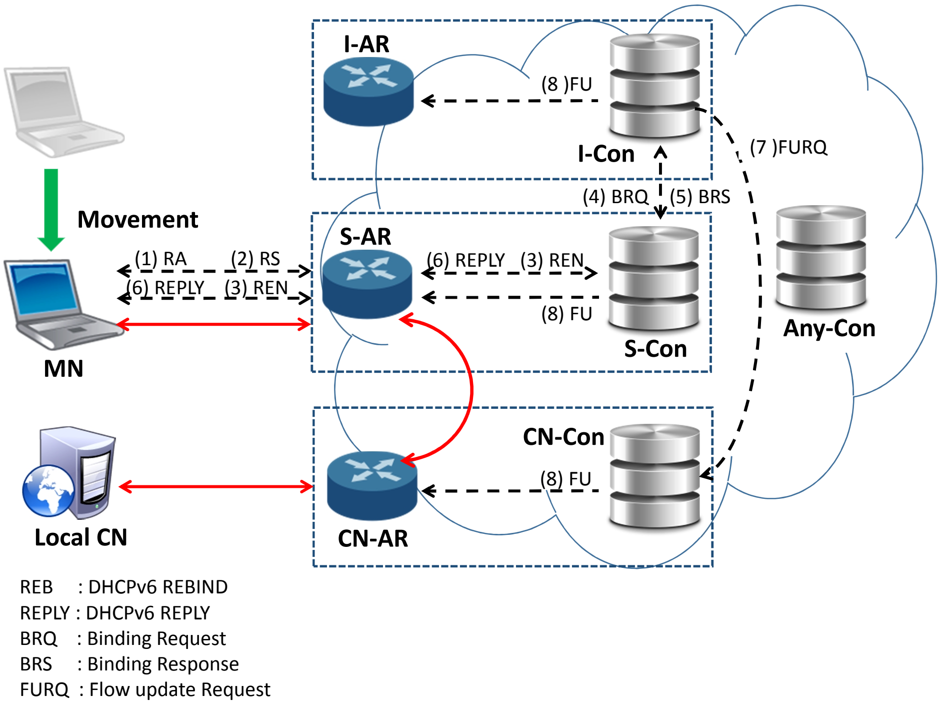


Figure 6.11: Handover and Flow Continuity.

6.5.3.2 Handover and session continuity

Figure 6.11 elaborates the signal flow during handover and the data flow after the successful handover. DMMSDN assumes that the reattachment of the MN followed by RS/RA exchange leads the MN to recognize that it has only gone through a temporary detachment from the ISP network. Thus, the MN will send the *REBIND* message attempting to reuse the same IPv6 address it was assigned previously, which is relayed by the S-AR to the S-Con. However, even if the MN sends the *SOLICIT* message in respond, failing to identify its reattachment, the S-Con will fetch the previously assigned IPv6 address while checking L2-IPv6 List following the procedure discussed in Sect.6.5.3.1. There is a nother exceptional case. The MN can respond with *CONFIRM* message in order to determine whether the IPv6 address is still appropriat to the link. It is also handled in the sameway as *REBIND* message.

When the *REBIND* or the *CONFIRM* message is received by the S-Con, it inspects the requested IPv6 address and identifies the I-Con which serves the MN. Then, it informs the the I-Con of MN's current location using the *Binding Request* message, where the I-Con can respond positively or negatively(Fig.6.12(1)) Negative respond will lead to initialize the connection of the MN again according to the DHCPv6 specification after the S-Con receives the negative *Binding Response* message. If the I-Con accepts the binding, then it updates the BCE (Fig.6.12(1)) accordingly and triggers flow table up-

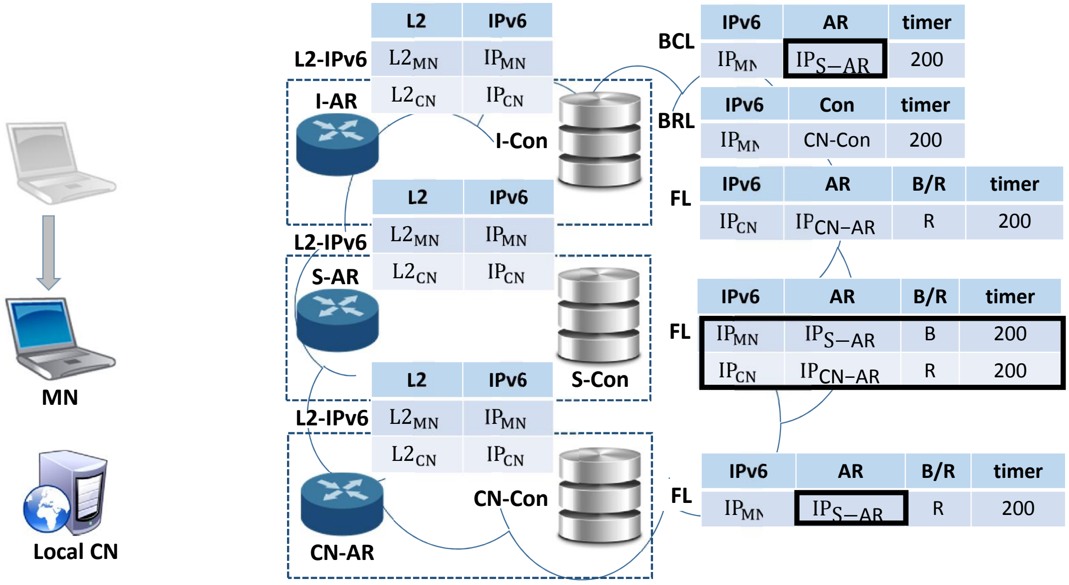


Figure 6.12: Memory Involved During Handover.

dates for all the entries in the BRL while responding the S-Con with the positive *Binding Response* message. At its reception, the S-Con updates its FL with MN‘s information indicating that it is a *binding* entry as indicated in Fig.6.12 (2). Positively affected by the flow table updates (Fig.6.12(3)), the data communication between the CN and the MN will continue and the optimal path between S-AR and the CN-AR will be taken by the subsequent packets.

6.5.3.3 De-registration

De-registration defined in DMMSDN can be of two types; (1) Graceful de-registration (Fig.6.13) and (2) timeout de-registration (Figs 6.14-6.17). Graceful de-registration occurs when the MN attempts to close the connection relinquishing the IPv6 address it was assigned and terminate the lease. This scenario is depicted in Fig.6.13. The *DHCPRELEASE* message (Fig.6.13(1)) is relayed by the currently serving AR of the MN (S-AR) to the S-Con. Thus, the S-Con attempts to withdraw the binding from the I-Con using the *Binding Release Request* message (Fig.6.13(2)), where the I-Con responds with an optional *Binding Release Respond* message (Fig.6.13(3)) while updating the BCE. Resultant consequences might lead the I-Con to release the L2-IPv6 registration by informing the SDN-controller set (Fig.6.13(5)). Nevertheless, I-Con should perform the mandatory flow table update for all the entries in the BRL (Fig.6.13(6)).

On the other hand, the timeout de-registration can occur for BCE, BRE, or FLE. BCE timeout and BRE timeout are elaborated in Figs 6.14 and 6.15, respectively. The

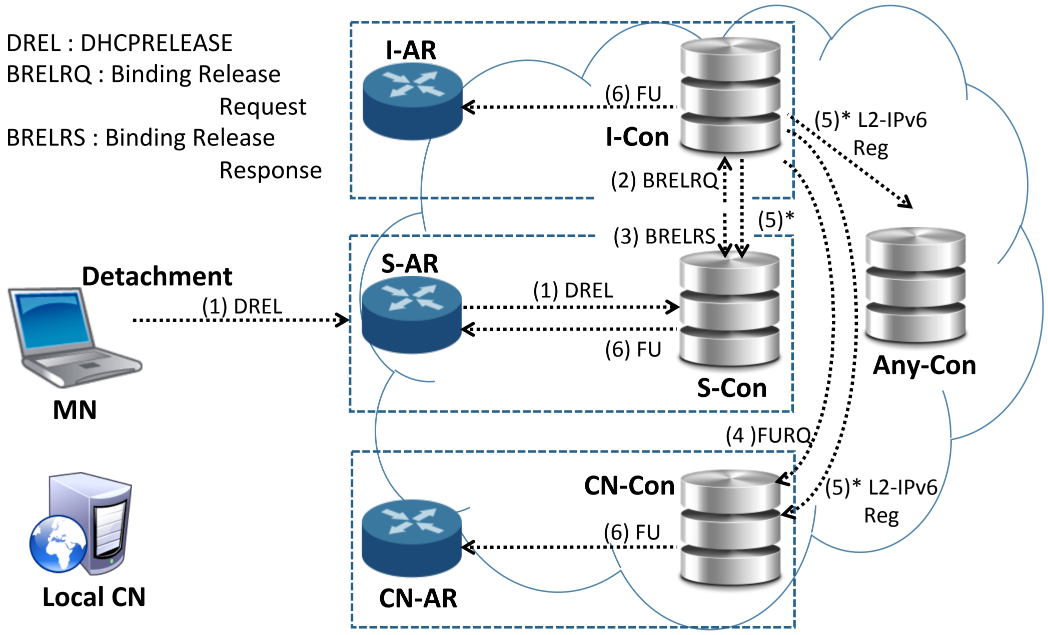


Figure 6.13: Graceful De-Registration.

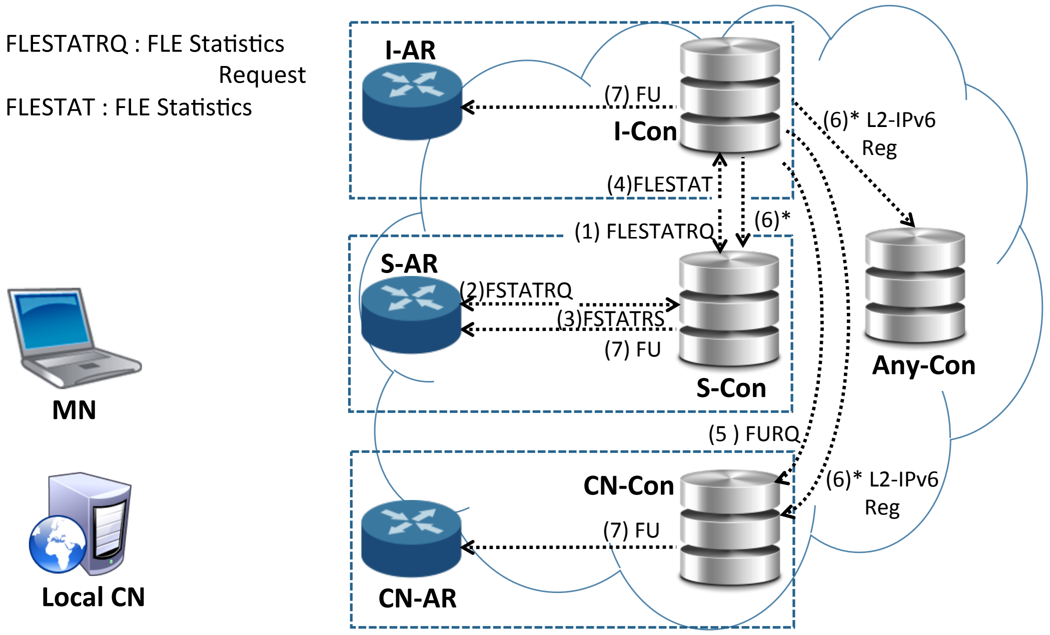


Figure 6.14: Occurrence of a BCE Timeout in I-Con.

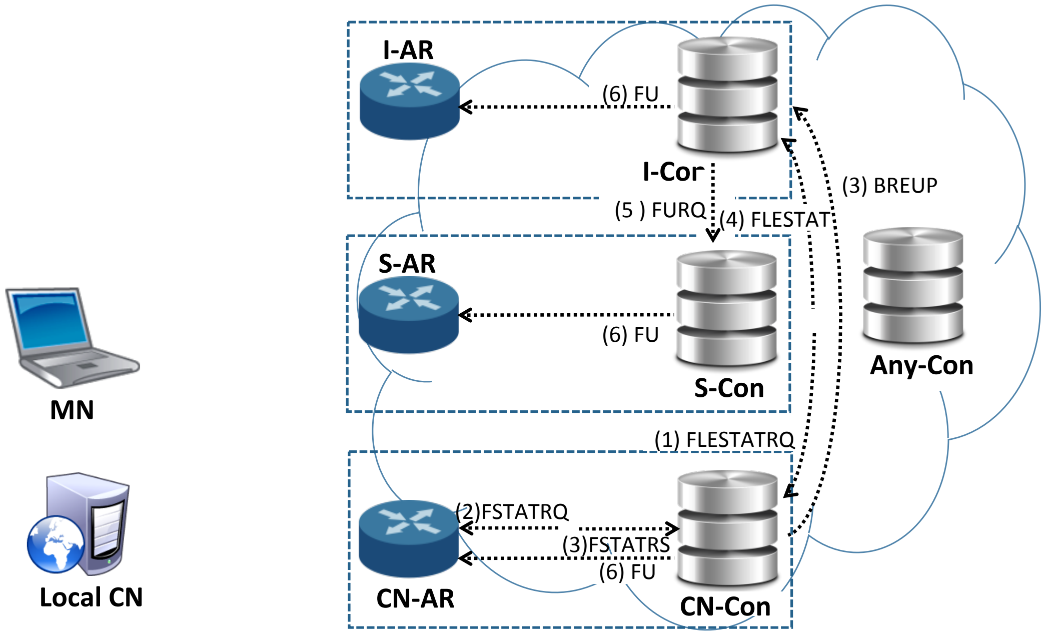


Figure 6.15: Occurrence of BRE Timeout in I-Con.

I-Con inquires the relevant SDN-controller (the S-Con for the BCE as shown in Fig.6.14 and the CN-Con as shown in Fig.6.15 or Any-Con for BRE timeout) for FLE statistics using the *FLE Statistics Request* message (Fig.6.14(1) / Fig.6.15(1)). Then, the SDN-controller inquires the respective AR using the *Flow Statistic Request* message (Fig.6.14(2) / Fig.6.15(2)) to which the AR will respond with the *Flow Statistics* message (Fig.6.14(3) / Fig.6.15(3)) informing whether the flow is active. It will be then reported to the I-Con by the receptive SDN-controller using the *FLE Statistics* message (Fig.6.14(4) / Fig.6.15(4)). If the timeout is verified, then the I-Con updates the BCE and BRE and triggers the required flow table updates (Fig.6.14(5)(7) / Fig.6.15(6)). However, if the timeout is ruled out due to active sessions, The Timer fields of the BCE and the BRE are simply updated.

FLE timeout can take place in the S-Con, the CN-Con, or Any-Con where the former two scenarios are shown in Figs.6.16 and 6.17. Latter scenario is exactly the same as the FLE timeout of the CN-Con. When the FLE timeout occurs, first the SDN-controller checks with respective AR whether the flow is active similar to the BCE and BRE timeouts discussed above (Fig.6.16(1)(2) / Fig.6.17(1)(2)). After the confirmation of the timeout, the SDN-controller (the S-Con, the CN-Con, or Any-Con) will send the *BCE Update* message (if the subjective FLE resides in S-Con: Fig.6.16(3)) or *BRE Update* message (if the subjective FLE resides in CN-Con or Any-Con: Fig.6.17(3)) as depicted in Figs.6.16 and 6.17, trying to withdraw the BCE and BRL entries from the I-Con. After the reception of the withdrawal message, the I-Con triggers flow table updates for the effected entries in the BRE (Figs.6.16(4)(5) / Figs.6.17(4)(5)).

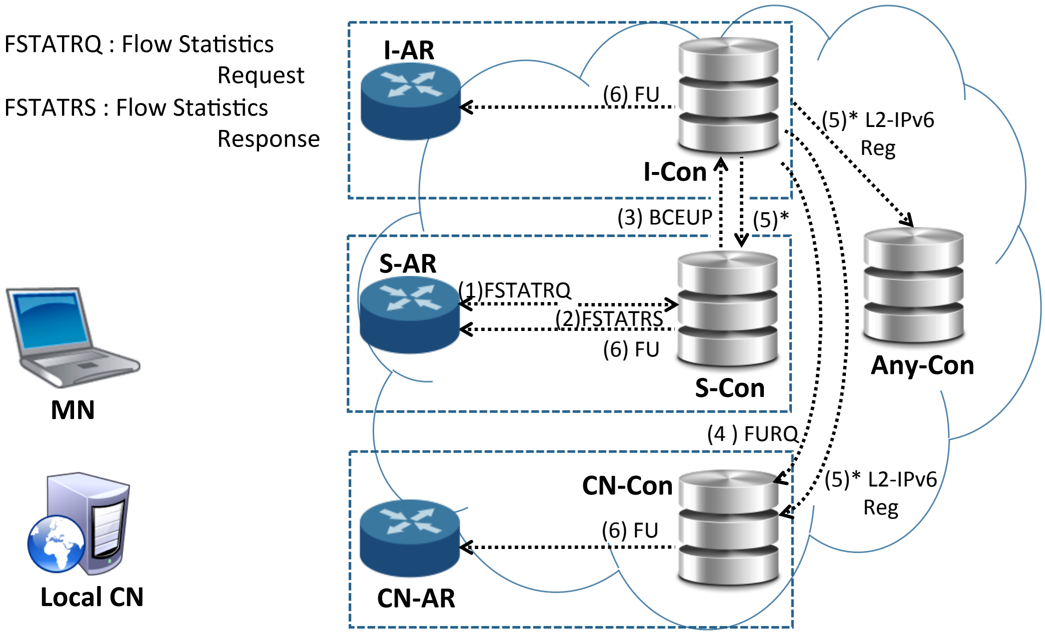


Figure 6.16: Occurrence of a FLE Timeout in S-Con.

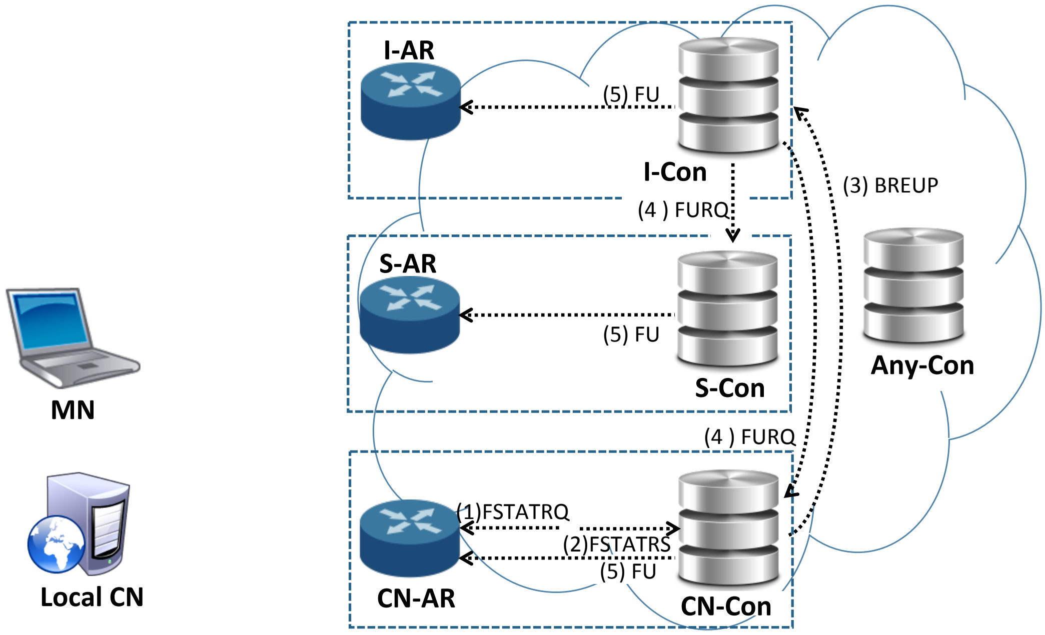


Figure 6.17: Occurrence of FLE Timeout in CN-Con.

Table 6.4: Parameters Used in MME Placement.

Evaluation	Synthetic Topology	Mapped Topology
Comparison	-	Degree and closeness based
Multiple SDN-controller	Degree and control based *x,y* coordinate based	Degree and closeness based Geolocation based

6.6 Performance Confirmation

Performance of DMMSDN is examined for two criteria. First, the feasibility of DMMSDN is assessed. In order to determine the fitness of DMMSDN, it is required to be compared with some DMM schemes that are already standardized or proposed. Thus, PMIP is selected as one of the referential schemes. Then, in order to compare the applicability, Distributed PMIP and RO-SDN are also selected as referential schemes. Selection of Distributed PMIP is due to the fitness it showed in the control-plane during the previous evaluation given in Chapt. 5. One reason to select RO-SDN is the performance it assured in the data-plane. Another reason to select RO-SDN is that it also considered SDN as its target network paradigm. Thus, the results can be used in order to determine the fitness of DMMSDN in the SDN environment as well.

As the second evaluation, the optimal SDN-controller distribution is determined. As already observed in the evaluation in Chapt. 4, different ISPs react to DMM schemes in different ways. Thus, it is vital to determine the economical design of DMMSDN for each ISP network. In order to generalize the optimization, tiers are used as the primary categorization of the ISPs. Then, the most appropriate distribution of SDN-controllers are determined for each of these categories in general.

6.6.1 Methodology

The control and data-planes of the considered referential are mathematically modeled as sub-emulators to be employed in SimNetDMM discussed in Chapt. 3. Then, the MME locator generator is tuned for degree and closeness-centrality based selection, and location-based selection. This enables multiple selection of MMEs using realistic areal criterion. Table 6.4 shows the overview of the MME selections employed during the two evaluations carried out.

For the performance comparison between the selected schemes, 50 locations to which 50 MN relocates are randomly selected out of the ARs predefined by the Rocketfuel data set. Then, 50 CNs are also selected from the predefined set of ARs and Gateways (GWs) which peer the ISP network with external networks. Thus, each handover scenario out of (50 MNs × 50 secondary locations) is considered to generate 50 × 50 × 50 test scenarios when each handover scenario is tested for 50 different sessions with the derived set of CNs.

For the multiple SDN-controller performance evaluation, 2,500 random test scenarios are derived as described below for each ISP topology considered. First, 50 random initial locations and visited locations through which the MN attaches to the network are selected. For each of these pairs, 50 random CNs are determined with which the MN establishes *sessions*. Thus, one scenario consists of one initial MN location, one random location to which it relocates, and a random CN with which the MN communicates continuously.

The BRITE topology generator only provides the distance and latency between routers, whereas the Rocketfuel data set provides latency and link weight information of the edges connecting routers. Thus, the common measurement which can be used in evaluating the performance is determined as the latency.

6.6.2 Results

6.6.2.1 DMMSDN Performance Comparison

The control and data-plane performances during the initialization of MN are shown in Figs. 6.18 and 6.19, respectively. They show CCDFs of control and data-plane performance.

The next figure, Fig. 6.20, depicts CCDFs while performing handover. Then, Fig. 6.21 shows the data delivery after the completion of successful handover. A comprehensive numerical summary of cost and latency of the each scheme for 95% and 100% test scenarios is given in Tables 6.5 and 6.6. This helps to obtain a marginal worse case performance comparison of the selected schemes. The field *sum* under initial and handover categories in Table 6.5 and Table 6.6 sums the time it takes to deliver the first packet after the initiation or handover. Thus, it can be considered as a representative measurement to distinguish the performance of the schemes.

When the control-plane during the initial attachment of the MN is considered, DMMSDN maintains better latency and cost in comparison to the rest. As far as the control-plane performance during the handover is considered, 90%-95% of the test scenarios have yielded the best control-plane performance with DMMSDN, but, the rest 5%-10% also assure the completion of handover within 80 ms with DMMSDN, which seconds only to DPMIP. In the data-plane, DMMSDN achieves optimal routing, whereas RO-SDN also manages to achieve optimal routing after the routing optimization procedure (Fig. 6.21 : *RO-SDN_later_pkts*). However, until that RO-SDN has the same results as the DPMIP (Fig. 6.21 :*RO-SDN_initial_pkts*), whereas DPMIP shows the worst results.

6.6.2.2 Optimal SDN-controller Determination

Figures 6.22 through 6.24 show the latency for initialization and population of flow tables of the synthetic topologies with 1,500, 750, and 250 routers. The set of figures, Figs. 6.25 to 6.27 depict the latency occurred while handling the handover of the same set of topologies.

The results obtained for the largest synthetic topology with 1,500 routers given in Fig.6.22 show a significant difference between the considered SDN-controller installments during initialization whereas five SDN-controller installment seems to outperform the other SDN-controller installments. Single SDN-controller installment performs the worst whereas the two SDN-controller installment seems to perform slightly better than the single SDN-controller installment. 25 SDN-controller installment yields

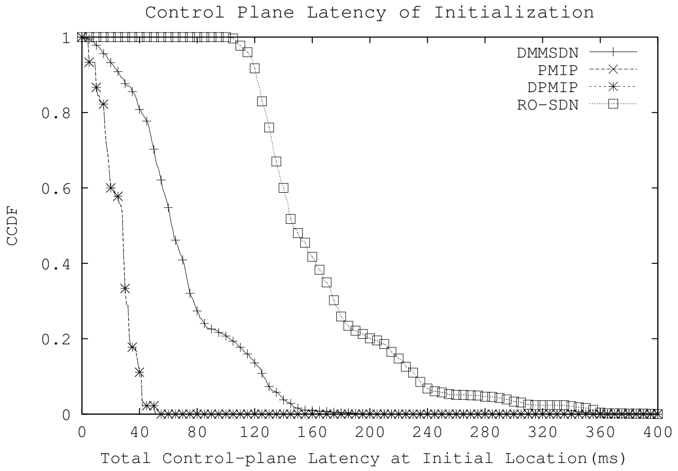


Figure 6.18: Initial Control-plane Latency Comparison of Schemes.

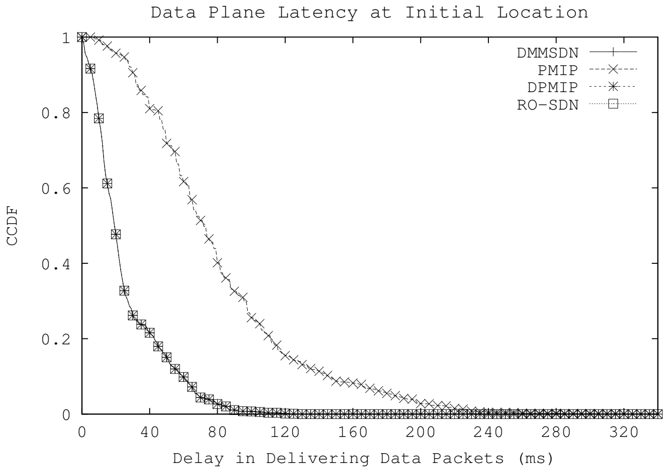


Figure 6.19: Initial Data-plane Latency Comparison of Schemes.

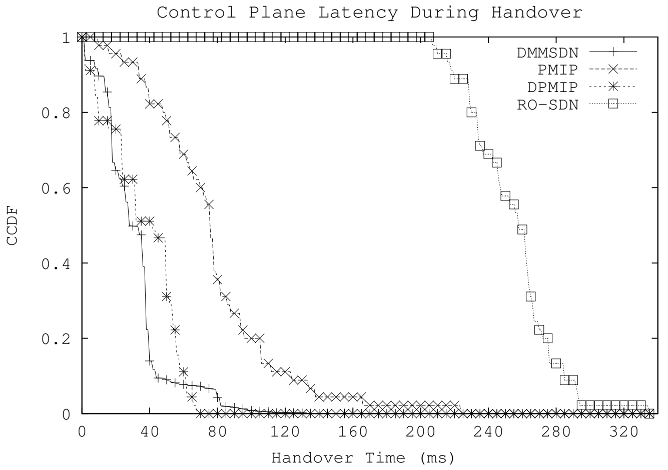


Figure 6.20: Control-plane Latency During Handover.

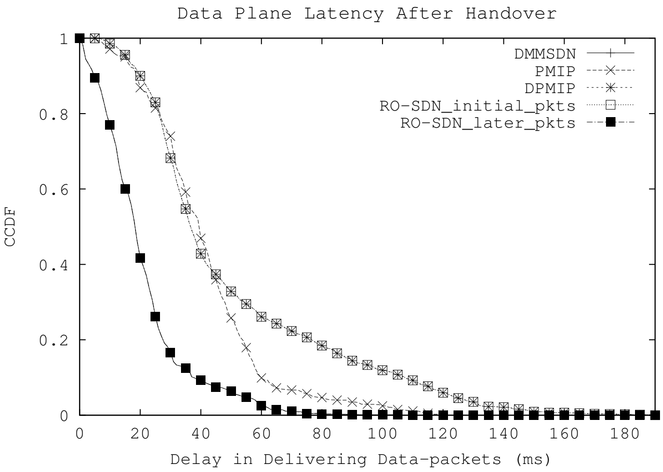


Figure 6.21: Data-plane Latency After Successful Handover.

Table 6.5: Summary of Latency Results Yielded for CCDF = 0.05 (95%).

	Initial Latency (ms)			Handover Latency (ms)		
	c-plane	d-plane	sum	c-plane	d-plane	sum
PMIP	42	184	226	138	77	215
DPMIP	42	69	111	62	124	186
RO-SDN	269	69	338	294	124	428
DMMSDN	137	69	205	79	54	133

slightly degraded results compared to the five SDN-controller installment. On the other hand, it shows mixed results during the handover. Even though the single SDN-controller installment seems to perform slightly better than the other three when the graphical representation in Fig.6.25 is considered, the difference is insignificant. When the overall results are considered, both 5 and 25 SDN-controller installments can generally viewed as candidate installments.

Topologies with 750 and 250 routers show mixed results in general unlike the previously discussed topology with 1,500 routers. Figure 6.23 shows a favorable results with 25 SDN-controller installment for the topology with 750 routers during initialization. However, looking at the descriptive statistics, it can be determined that the five SDN-controller installment yields least worst latency, whereas the other statistics remain the same. During the handover, the latency curve in Fig.6.23 shows better performance regardless of having slightly higher worst latency value which is comparatively insignificant.

Finally, the smallest synthetic topology with 250 routers does not show a significant difference between results gained during different SDN-controller installments. However, it is clear that the single SDN-controller installment shows the worst performance during initialization as well as while performing the handover. After graphical and tabular comparison, two SDN-controller installment can be identified as the best distribution scenario.

Figures 6.28 through 6.31 and Figs. 6.32 through 6.35 show latency based per-

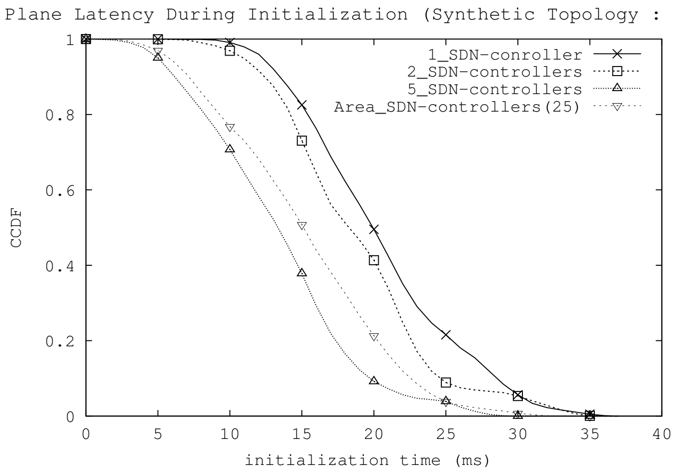


Figure 6.22: Initial Control-plane Latency : Synthetic Topology (1500 Routers).

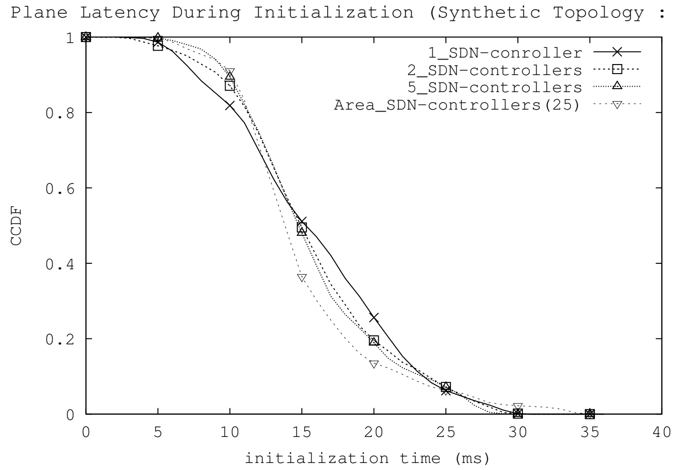


Figure 6.23: Initial Control-plane Latency : Synthetic Topology (750 Routers).

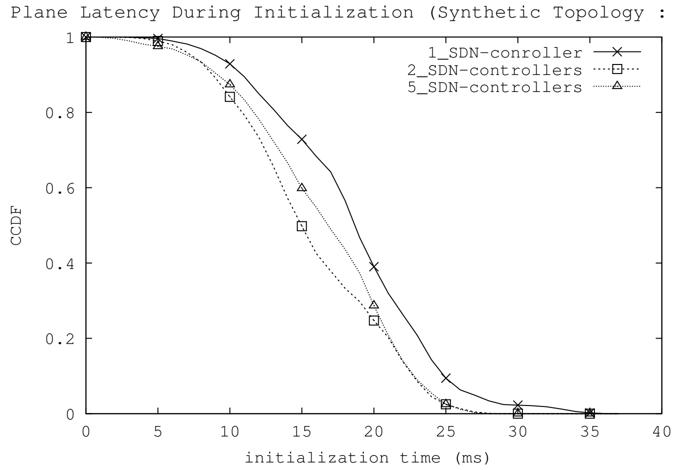


Figure 6.24: Initial Control-plane Latency : Synthetic Topology (250 Routers).

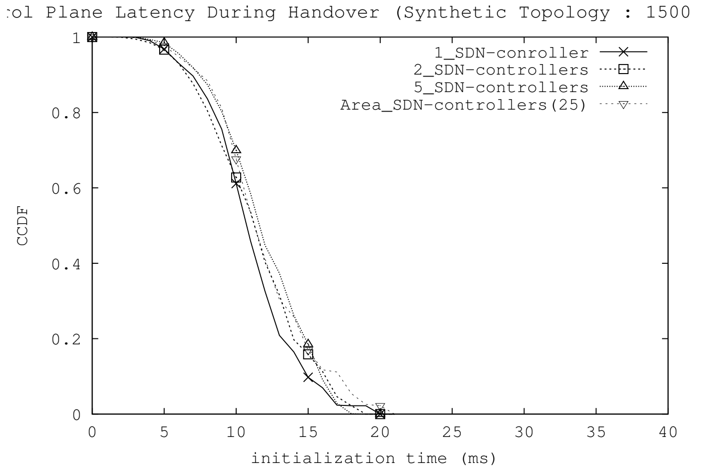


Figure 6.25: Control-plane Latency During Handover : Synthetic Topology (1500 Routers).

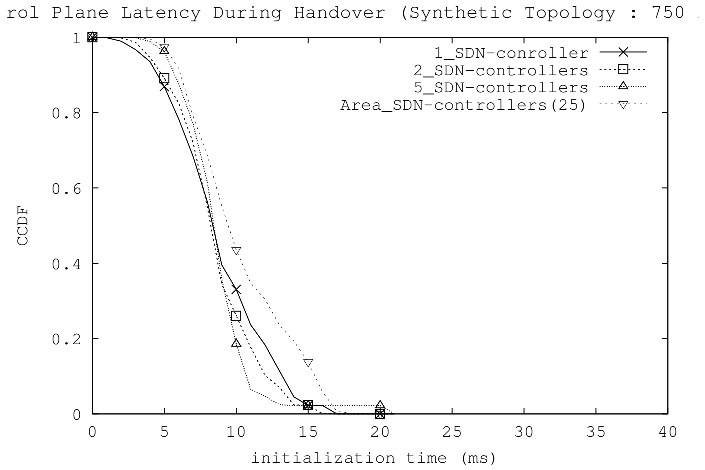


Figure 6.26: Control-plane Latency During Handover : Synthetic Topology (750 Routers).

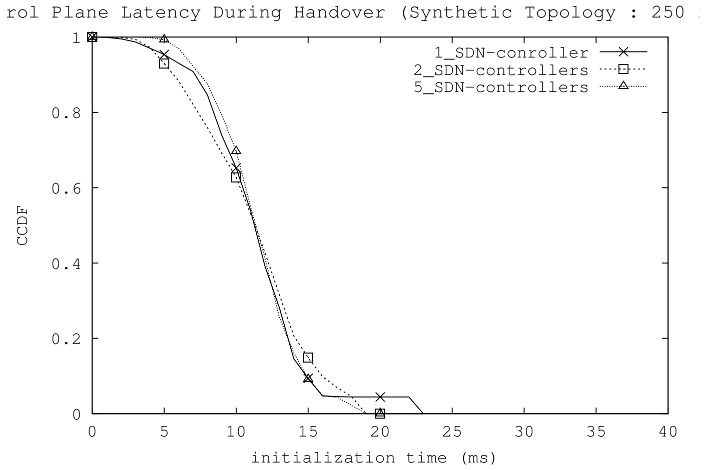


Figure 6.27: Control-plane Latency During Handover : Synthetic Topology (250 Routers).

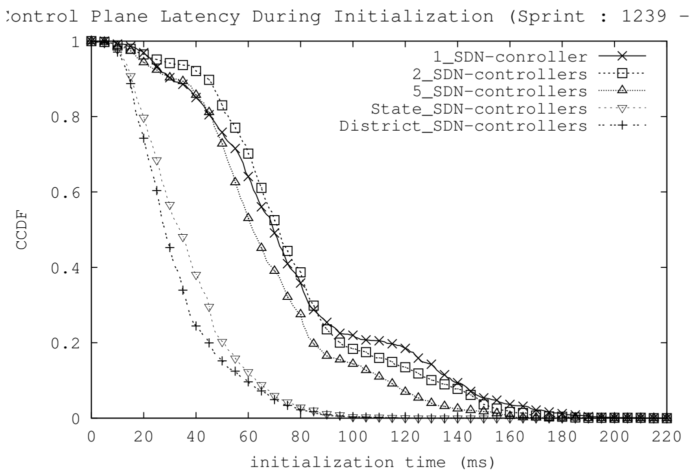


Figure 6.28: Initial Control-plane Latency Sprint (ASN:1239).

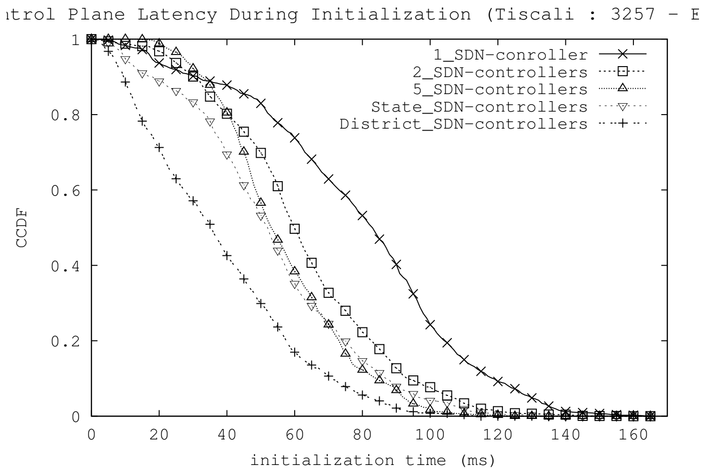


Figure 6.29: Initial Control-plane Latency Tiscali (ASN:3257).

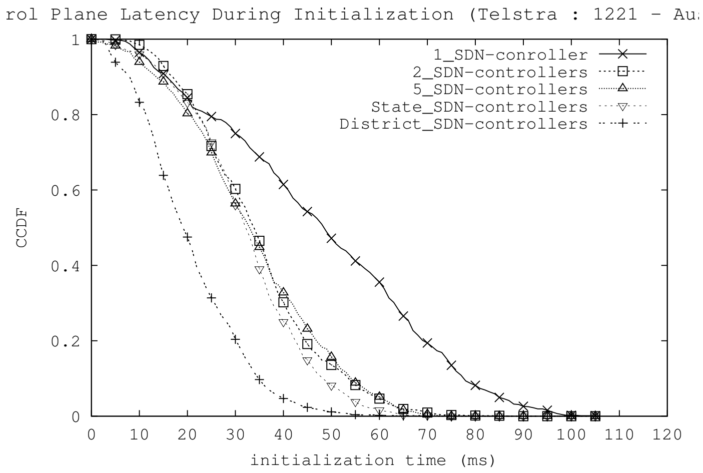


Figure 6.30: Initial Control-plane Latency Testra (ASN:1221).

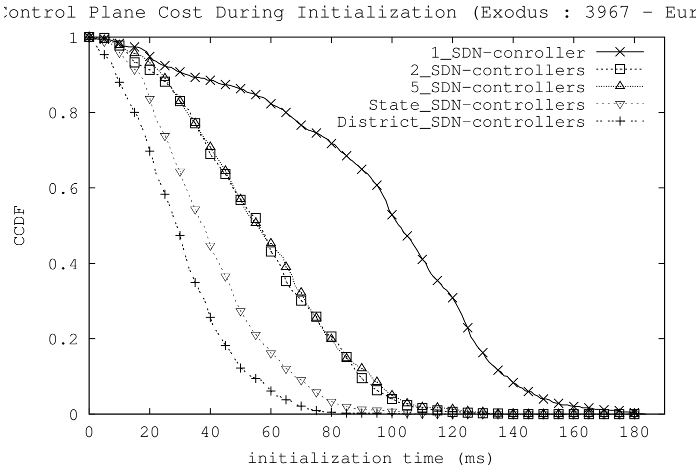


Figure 6.31: Initial Control-plane Latency (ASN:3967).

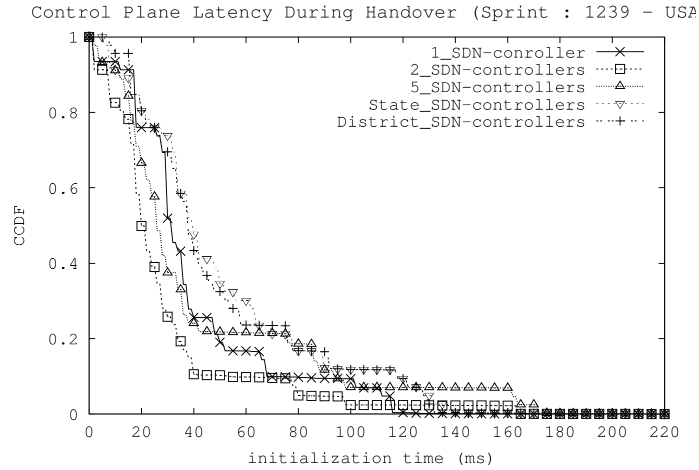


Figure 6.32: Control-plane Latency During Handover Sprint (ASN:1239).

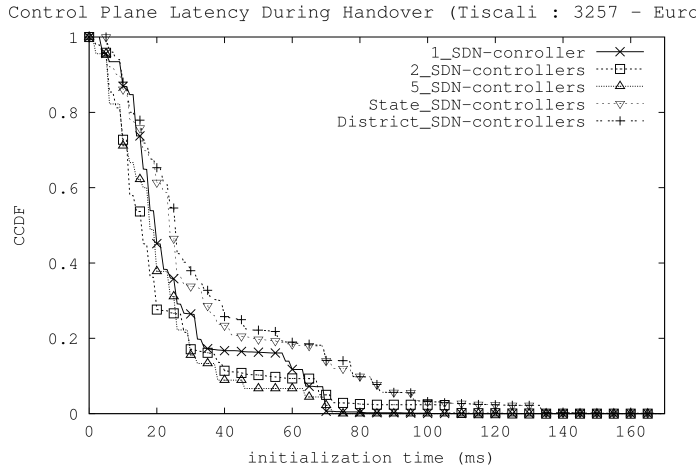


Figure 6.33: Control-plane Latency During Handover Tiscali (ASN:3257).

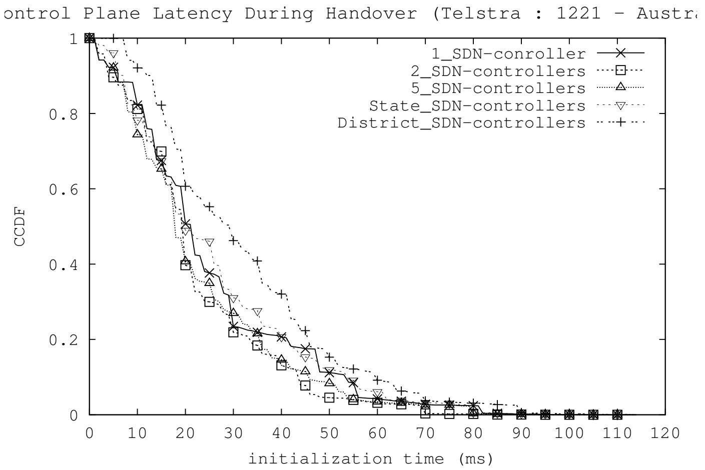


Figure 6.34: Control-plane Latency During Handover Testra (ASN:1221).

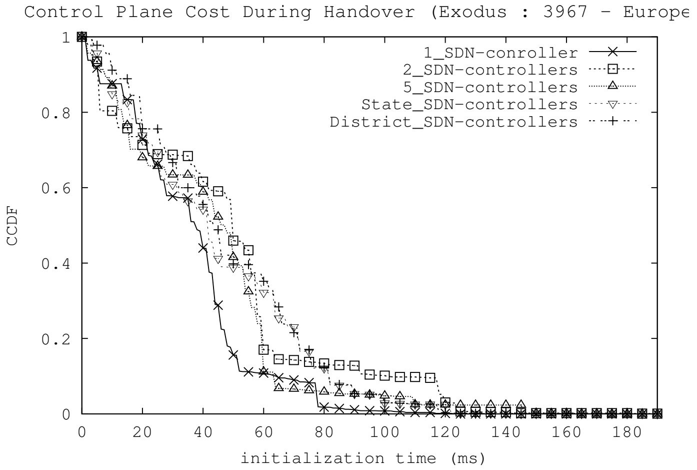


Figure 6.35: Control-plane Latency During Handover Exodus (ASN:3967).

Table 6.6: Summary of Latency Results Yielded for CCDF = 0.00 (100%).

	Initial Latency (ms)			Handover Latency (ms)		
	c-plane	d-plane	sum	c-plane	d-plane	sum
PMIP	51	338	389	224	122	326
DPMIP	51	126	177	68	190	258
RO-SDN	400	126	526	335	190	525
DMMSDN	203	126	329	162	106	268

formance observed during initialization and handover for the mapped ISP topologies: Sprint (ASN: 1239), Tiscali (ASN: 3257), Testra (ASN: 1221), and Exodus (ASN: 3967), respectively. In general, regardless of the topology (its scale, and the distribution), initialization yields considerably variant performances. Single SDN-controller installment shows the worst performance whereas district or city based SDN-controller installment shows the best performance. However, when the performance during the handover is considered, it does not show significant difference for varying SDN-controller

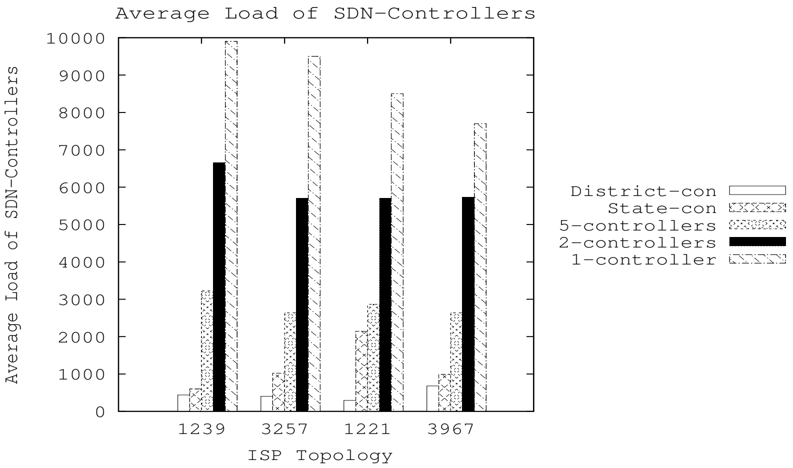


Figure 6.36: Average Event Load of SDN-controllers.

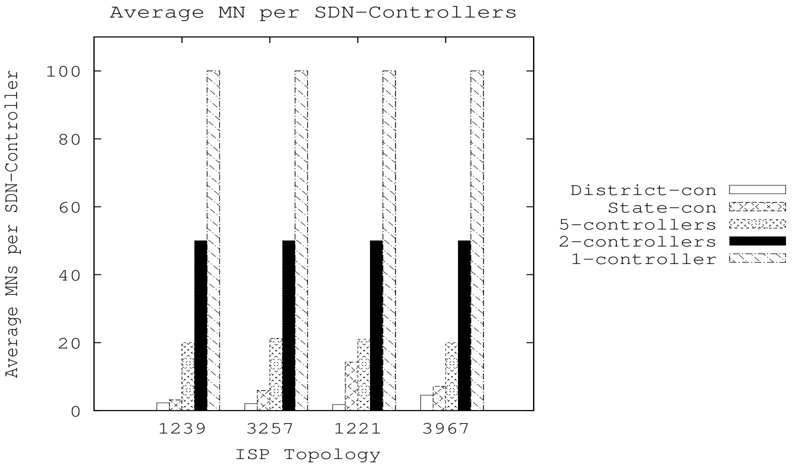


Figure 6.37: Average MNs per SDN-controller.

installments.

Being the largest ISP topology out of the selected, the Sprint:1239 topology seems to show much improved initialization performance with geolocation based SDN-controller installments. Two performance bands are visible. One includes degree and closeness based SDN-controller selection and another includes the geolocation based SDN-controller installment. Nonetheless, the handover performance is much similar regardless of the number of SDN-controllers employed. It can be seen that the overhead of controlling taken place during the handover does not significantly affect the overall performance in larger *tier-1* ISP topologies. For ISP networks which has a highly scattered topology resembling Sprint:1239 with significant amount of ARs and mostly concentrated on couple of continents, the district or city based SDN-controller installment can be recommended.

The tier-2 ISP topology, Tiscali:3257, also shows worst results with the single SDN-controller installment for initialization and the five SDN-controller and state SDN-controller installment seem to perform in the same manner. Interestingly, Tiscali:3257 seems to yield better results with five SDN-controller installment resulting in better latency in the worst case. Looking at the Tiscali:3257 topology, it covers Europe and some parts of north America in a highly scatted manner. However, it has a very limited number of ARs. This unique set of properties can be considered as a reason for showing unsustainable results when a large number of SDN-controllers are employed.

Thus, it can be identified that, for ISP topologies covering a large area including several continents but with least number of Points of Present (POPs) and ARs, having a small number of SDN-controllers is sufficient to obtained better performance.

Although both Teltra:1221 and Exodus:3967 networks belong to the *tier-3*, they display slightly different performance patterns during initialization. Teltra:1221 shows much worst results with the single SDN-controller installment whereas the worst cost performance observed during initialization for the single SDN-controller installment of Exodus:3967 is better than that of two and five SDN-controller installments. However, when the latency is concerned, both Testra:1221 and Exodus:3967 show three performance groups. Even though the latency pattern in Figs. 6.34 and 6.35 do not show clear performance differences, Tables 6.7 and 6.8 assist in determining the best performance for each ISP topology based on the best mean and the worst latencies observed. Thus, in general, the state based SDN-controller installment is suggested. When looking at the selected *tier-3* ISP topologies, they consists of a well established POP network, thus having a significant amount of ARs. However, Testra:1221 seems to have the most established and firm distribution of POPs geographically. On the other hand, although Exodus:3967 is a smaller ISP network, it covers few continents. It can be seen as a reason for the state based SDN-controller installment to perform comparatively better.

Figures 6.36 and 6.37 show the average event load and the number of MNs handled by SDN-controllers in all the mapped ISP topologies, respectively. A dramatic load reduction is seen as the number of SDN-controllers increase, despite of the resultant communication between SDN-controllers. The size of the additional mobility handling memory required per SDN-controller can be considered proportional to the number of MNs handled by the SDN-controller. Thus, the results confirm that the average memory space occupied in the SDN-controller decreases drastically when the degree of distribution increases. Thus, it confirms the capability of employing DMMSDN without introducing heavy control-plane load. In overall, the results confirms the applicability of DMMSDN in practical situations.

After closely analyzing the collective latencies of initialization and handling handover using both the graphical and tabular interpretations, the optimal SDN-controller installment is determined and highlighted in Tables 6.7 and 6.8.

6.7 Applicability and Limitations of DMMSDN

DMMSDN assumes a distributed SDN environment, whereas the local network is managed using multiple SDN-controllers. In the above evaluations, the inter SDN-controller communication that takes place is assumed follow the policies and latencies observed in the current IPv6 environment. However, in actual SDN environment, it is expected to be different. It can be considered as under-estimated in this study. Therefore, the result observed may vary. Specifically, the result can be expected to improve.

In the current SDN context, reactive flow table updates are assumed to take place most of the times. Nevertheless, the SDN architecture does not restrict the proactive

Table 6.7: Statistics of Observed Latencies - Initialization.

Topology Type	Topology / ASN	SDN-Con	Initialization Latency		
			Best Latency (ms)	Mean Latency (ms)	Worst Latency (ms)
Synthetic	1500	1	6	19	36
		2	5	18	35
		5	1	13	29
		25	1	15	32
	750	1	3	15	30
		2	1	14	30
		5	3	14	30
		25	3	13	35
	250	1	3	18	36
		2	2	14	28
		5	1	16	27
Mapped	1239	1	9	69	232
		2	1	71	210
		5	2	61	189
		state	1	33	120
		district	1	28	120
	3257	1	1	82	165
		2	1	59	155
		5	14	53	123
		state	1	52	136
		district	1	35	113
	1221	1	4	48	105
		2	5	33	186
		5	1	32	78
		state	1	32	73
		district	2	19	70
	3967	1	1	102	183
		2	4	55	137
		5	2	55	137
		state	1	37	115
		district	1	28	104

Table 6.8: Statistics of Observed Latencies - Handover.

Topology Type	Topology / ASN	SDN-Con Selection Approach	Handover Latency		
			Best Latency (ms)	Mean Latency (ms)	Worst Latency (ms)
Synthetic	1500	1	3	11	19
		2	2	11	18
		5	4	11	18
		25	3	11	20
	750	1	1	8	16
		2	2	8	16
		5	3	8	20
		25	4	9	19
	250	1	1	11	22
		2	1	11	19
		5	4	11	18
Mapped	1239	1	2	31	166
		2	2	19	211
		5	2	25	232
		state	6	39	214
		district	8	37	214
	3257	1	4	19	110
		2	2	15	134
		5	2	17	71
		state	6	23	168
		district	1	25	168
	1221	1	2	21	105
		2	2	19	93
		5	2	17	103
		state	2	19	103
		district	2	29	113
	3967	1	2	37	122
		2	2	49	194
		5	2	47	173
		state	4	41	185
		district	4	43	181

flow table updates. As the proactive flow table updates may introduce significant processing times, it may degrade the performance of DMMSDN. In this study, the processing times are neglected. However, it should be noted that, when compared to the communication times, the extension of the processing time might be extremely insignificant.

Most implementation efforts of SDN paradigm seen currently requires the AR to forward initial packets of all the flows to the central SDN-controller. Then the SDN-controller determines the how the flow should be treated. This introduces an unnecessary control overhead for non-mobile nodes. The address blocking introduced by DMMSDN can be used to exempt this situation. The mobility disabled IPv6 address block can be aggregated and the ARs and SDN switches in the core can be pre configured to handle the flows of stationary nodes as general IPv6 flows. This can be used to reduce the initialization delay to a greater extend. The above evaluation does not consider stationary nodes.

6.8 Scalability and Fault Tolerance

The evaluation described in this paper neglects the searching , residence, and processing times at all the SDN-nodes. Further, it does not discuss the overheads due to fault tolerant measurements. Even though the resident and the processing times depend on the performance of underlying equipment, the searching time and the fault-tolerance is partially dependent on the DMMSDN design. Thus, a sustainable memory and fault-tolerance design are vital for an economical implementation.

6.8.1 Required Memory Size

In general, SDN architecture simply suggests a logically centralized SDN-controller. Physically, the SDN-controller can be decentralized in distributed set of machines. Thus, it may require to install redundant MME's in physical SDN-controllers to enable MM. This may lead to excessive usage of memory space and inconsistency. DMMSDN suggests to overcome this issue by minimizing the redundancy of mobility information maintained by each SDN-controller while ensuring consistency. The total memory required by the DMMSDN MMEs is dependent on the number of subscribers per area network.

The total space occupied in each SDN-controller due to mobility handling is mathematically derived as follows. Let MN_{ij} be the i^{th} MN, which belongs to the j^{th} area network. m stands for the number SDN-controllers (the number of area networks) in the ISP. N is the total number of subscribers in the ISP network, whereas n_j and n_k are the number of subscribers in the j^{th} and k^{th} area networks, respectively. s_k^{ij} and r_k^{ij} represent Boolean values, whereas the former is 1 if there is at least one sessions from the SDN-controller of the k^{th} area network for MN^{ij}. Similarly, r_k^{ij} is set to 1 if there exists any update request regarding MN^{ij} from the SDN-controller managing k^{th} area network. X_{size} symbolizes the size of the Entry X.

The memory occupied by the L2-IPv6 List in all the SDN-controllers, $L2IPv6_{size}$

can be given by the formula

$$L2IPv6_{size} = \sum_{k=0}^{m}\{n_k \times (L2_{size} + IPv6_{size})\}. \tag{6.1}$$

The space taken by FLs in any SDN-controller j, FLE_{jsize} can be computed as follows.

$$FLE_{jsize} = \sum_{k=0}^{m}\sum_{i=0}^{n_j}\{(s_k^{ij} \| r_k^{ij}) \times (IPv6_{size} + IPv6_{size} + B/R_{size} + timer_{size})\}. \tag{6.2}$$

Further, the additional memory required by BCL and BRL in the I-Con j are given by BCL_{jsize} and BRL_{jsize} can be defined as

$$BCL_{jsize} = \sum_{k=0}^{m}\{n_k \times (IPv6_{size} + AR_{size} + timer_{size})\} \tag{6.3}$$

and

$$BRL_{jsize} = \sum_{k=0}^{m}\sum_{i=0}^{n_j}\{(s_k^{ij} \| r_k^{ij}) \times (IPv6_{size} + Con_{size} + timer_{size})\}. \tag{6.4}$$

In order to minimize the searching time occurring in the MMEs, this thesis suggests following memory design, envisioning an economical implementation. First, common memory allocations are addressed. In the L2-IPv6 List, the search will be based on the L2 address, thus using a hash table or a skip list will reduce the time it takes to skim through the list which has a large non-persistent range. The insert, search, and delete operations have the average time complexity of O(1) and the worst-case time complexity of O(N). For skip list, time complexities for the average and worst cases to perform the above three operations are O(log N) and O(N), respectively. The FLs can be considered as an extension to the forwarding table employed in the SDN-controller, thus the data structure utilized to define flow rules can be considered as the candidate. However, hash table can be beneficial for faster search since FLEs may include IPv6 addresses which belongs to the non-persistent global range. That is the IPv6 addresses of FLEs are arbitrary due to the random nature of sessions.

For the BCL and the BRL in I-Con, a simple Binary Search Tree (BST) based implementation is a better suiting suggestion, where the entries are sorted based on the IPv6 address. A self-balanced BST will be a better option, where it improves the searching time. The BCL will only include IPv6 addresses that belong to the IPv6 address range of the area network and the size will be bounded to the number of subscribers. On the other hand, as only one entry per SDN-controller is inserted for any MN in the BRL, the BRL will include only a limited number of entries per MN, which is limited to the number of area networks in the ISP. Further, the IPv6 address range is expected to be much persistent due to prioritized and dynamic allocations. Thus, a fully sorted data structure better suits the BCL and the BRL. Search time complexity in the any SDN-controller, for both the worst and average cases are O(log N) with self-balanced BST.

6.8.2 Fault Tolerance

Common failures that might occur during the mobility handling can be listed as follow.

Table 6.9: Memory Size of BCL in Bytes (BCE_{size}=34Bytes).

SUBs per SDN-con.	Area Networks (ANs)	Sessions with MN per AN		
		0	10^1	10^2
10^3	0.5×10^2	34×10^3 (33 KB)	34×10^3 (33 KB)	34×10^3 (33 KB)
	10^2	34×10^3 (33 KB)	34×10^3 (33 KB)	34×10^3 (33 KB)
10^5	0.5×10^2	34×10^5 (3.2 MB)	34×10^5 (3.2 MB)	34×10^5 (3.2 MB)
	10^2	34×10^5 (3.2 MB)	34×10^5 (3.2 MB)	34×10^5 (3.2 MB)
10^7	0.5×10^2	34×10^7 (324 MB)	34×10^7 (324 MB)	34×10^7 (324 MB)
	10^2	34×10^7 (324 MB)	34×10^7 (324 MB)	34×10^7 (324 MB)

Table 6.10: Memory Size of BRL in Bytes (BRE_{size} =19Bytes).

SUBs per SDN-con.	Area Networks (ANs)	Sessions with MN per AN		
		0	10^1	10^2
10^3	0.5×10^2	- -	$19 \times 0.5 \times 10^2 \times 10^3$ (927 KB)	$19 \times 0.5 \times 10^2 \times 10^3$ (927 KB)
	10^2	- -	$19 \times 10^2 \times 10^3$ (1.8 MB)	$19 \times 10^2 \times 10^3$ (1.8 MB)
10^5	0.5×10^2	- -	$19 \times 0.5 \times 10^2 \times 10^5$ (90 MB)	$19 \times 0.5 \times 10^2 \times 10^5$ (90 MB)
	10^2	- -	$19 \times 10^2 \times 10^5$ (181 MB)	$19 \times 10^2 \times 10^5$ (181 MB)
10^7	0.5×10^2	- -	$19 \times 0.5 \times 10^2 \times 10^7$ (8.8 GB)	$19 \times 0.5 \times 10^2 \times 10^7$ (8.8 GB)
	10^2	- -	$19 \times 10^2 \times 10^7$ (17.7 GB)	$19 \times 10^2 \times 10^7$ (17.7 GB)

- Failure of MME
  - Functional Failure
  - Data Failure
- MM information inconsistency (lack of convergence)

In general, DMMSDN avoids a single point of failure by eradicating a centralized component handling mobility. However, it attempts to achieve scalability and consistency by avoiding redundant binding information maintenance. Thus, it emphasizes

Table 6.11: Memory Size oL2IPv6 List in Bytes ($L2IPv6\_Entry_{size}$=22Bytes).

SUBs per SDN-con.	Area Networks (ANs)	Sessions with MN per AN		
		0	10^1	10^2
10^3	0.5×10^2	$22 \times 0.5 \times 10^2 \times 10^3$ (1 MB)	$22 \times 0.5 \times 10^2 \times 10^3$ (1 MB)	$22 \times 0.5 \times 10^2 \times 10^3$ (1 MB)
	10^2	$22 \times 10^2 \times 10^3$ (2 MB)	$22 \times 10^2 \times 10^3$ (2 MB)	$22 \times 10^2 \times 10^3$ (2 MB)
10^5	0.5×10^2	$22 \times 0.5 \times 10^2 \times 10^5$ (105 MB)	$22 \times 0.5 \times 10^2 \times 10^5$ (105 MB)	$22 \times 0.5 \times 10^2 \times 10^5$ (105 MB)
	10^2	$22 \times 10^2 \times 10^5$ (209 MB)	$22 \times 10^2 \times 10^5$ (209 MB)	$22 \times 10^2 \times 10^5$ (209 MB)
10^7	0.5×10^2	$22 \times 0.5 \times 10^2 \times 10^7$ (10.2 GB)	$22 \times 0.5 \times 10^2 \times 10^7$ (10.2 GB)	$22 \times 0.5 \times 10^2 \times 10^7$ (10.2 GB)
	10^2	$22 \times 10^2 \times 10^7$ (20.5 GB)	$22 \times 10^2 \times 10^7$ (20.5 GB)	$22 \times 10^2 \times 10^7$ (20.5 GB)

Table 6.12: Memory Size of FL in Bytes (FLE_{size}=35Bytes).

SUBs per SDN-con.	Area Networks (ANs)	Sessions with MN per AN		
		0	10^1	10^2
10^3	0.5×10^2	- -	$35 \times 10^1 \times 10^3$ (34 KB)	$35 \times 10^2 \times 10^3$ (3.3 MB)
	10^2	- -	$35 \times 10^1 \times 10^3$ (34 KB)	$35 \times 10^2 \times 10^3$ (3.3 MB)
10^5	0.5×10^2	- -	$35 \times 10^1 \times 10^5$ (33 MB)	$35 \times 10^2 \times 10^5$ (333 MB)
	10^2	- -	$35 \times 10^1 \times 10^5$ (33 MB)	$35 \times 10^2 \times 10^5$ (333 MB)
10^7	0.5×10^2	- -	$35 \times 10^1 \times 10^7$ (3.2 GB)	$35 \times 10^2 \times 10^7$ (33 GB)
	10^2	- -	$35 \times 10^1 \times 10^7$ (3.2 GB)	$35 \times 10^2 \times 10^7$ (33 GB)

that the mobility information (binding) of a specific MN is stored only on a signal SDN-controller. Theoretically, in the SDN architecture, a failure of a SDN-controller triggers a failure in its area network. This leads the BCL and the BRL to create a single point of failure for a given IPv6 address range which follows an area network malfunctioning. In order to maintain the DMM of the SDN-controller's area unfailed during an occasion where an SDN-controller fails, requires the redundant storage of BCE and BRL. Maintaining backups of the BCE and the BRL can overcome such a failure. DMMSDN assumes that the transparent handling of such a failure will be facilitated by an underlying SDN fault tolerance mechanism. However, even without redundant memory for the BCL and the BRL as mentioned above, DMMSDN can be enhanced to handle such a failure if a different controller can take over the area network of a failed SDN-controller. Figure 6.38 depicts an example.

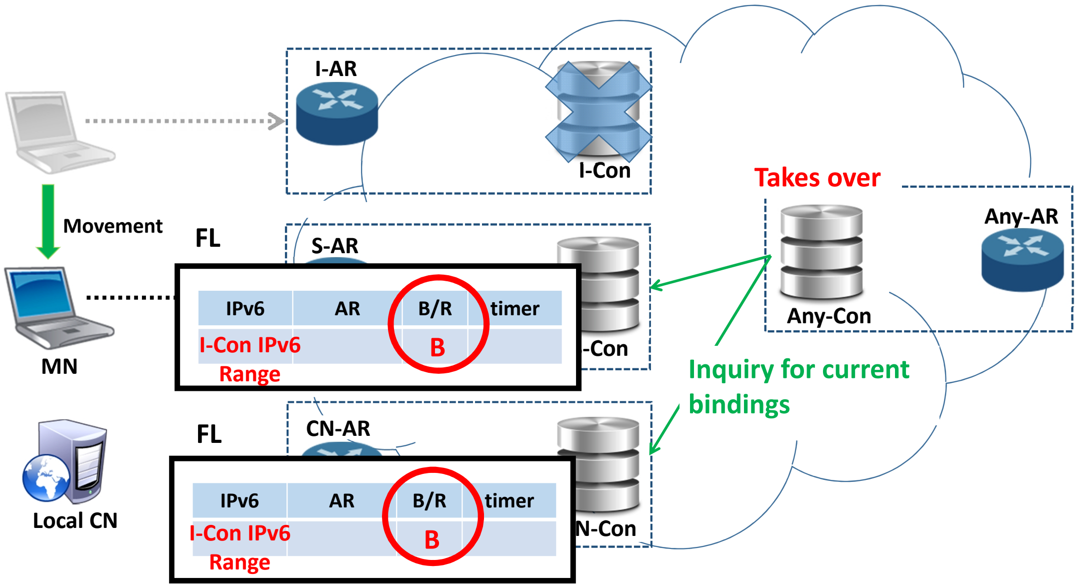


Figure 6.38: DMMSDN Fault Tolerance.

First, the SDN-controller which takes over the failed SDN-controller, inquires the FLE information from all the other SDN-controllers for the set of IPv6 addresses granted by the failed SDN-controller. Each SDN-controller can respond this inquiry using an *abstract response* mentioning its bindings or requests for the given IPv6 address set whereby the inquired SDN-controller can create and maintain the required BCL and BRL accordingly. Here, the most critical information is the binding information since it determines the reachability of the MN. The SDN-controller which took over the failed one can update the rest of the SDN-controllers about the expansion of the IPv6 address range it serves and it can continue to perform any flow update on behalf of the MNs belonging to the failed SDN-controller whenever required until the failed SDN-controller takes over again. When the failed SDN-controller manages to function again, first should inquire all the controllers to identify the SDN-controller which took over while its absence. Then, it should update all the records in the BCL and the BRL inquiring that SDN-controller. For better inconsistency handling, the failed SDN-controller does not claim the IPv6 address range until all the records are transferred. Finally, it should reclaim its IPv6 address range by informing all the SDN-controllers and requesting the SDN-controller which was managing the IPv6 address range during its failure to release it. Further, the SDN-controller which took over should inform any change took place in between the MM information transfer and the IPv6 address range release to the failed SDN-controller to avoid any inconsistency.

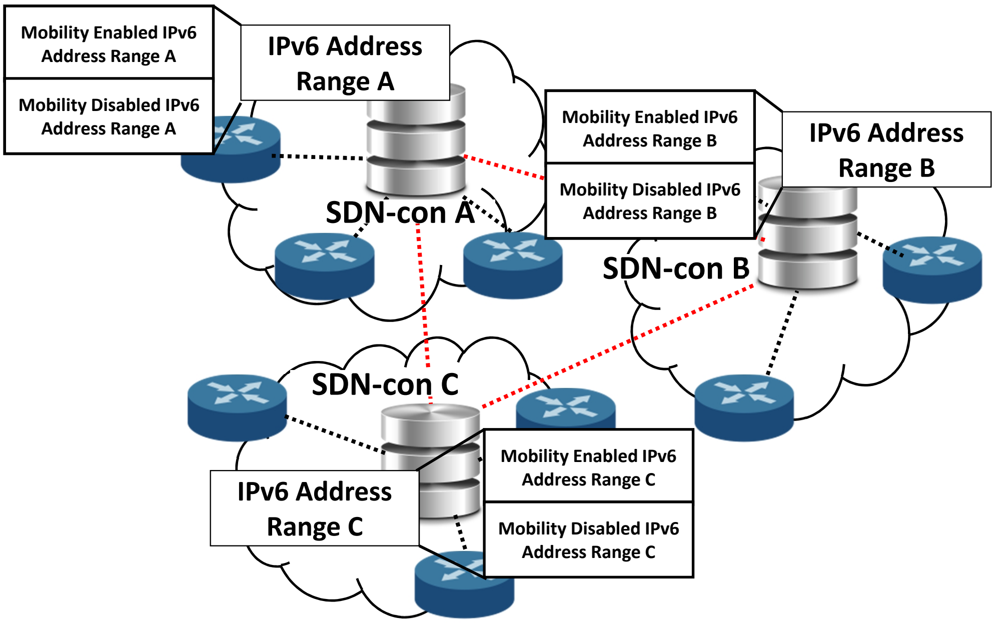

Mobility Enabled IPv6 Address Range A
Mobility Disabled IPv6 Address Range A
IPv6 Address Range A
SDN-con A
Mobility Enabled IPv6 Address Range B
Mobility Disabled IPv6 Address Range B
IPv6 Address Range B
SDN-con B
SDN-con C
IPv6 Address Range C
Mobility Enabled IPv6 Address Range C
Mobility Disabled IPv6 Address Range C

Figure 6.39: IPv6 Address Aggregation.

6.9 Reducing Initial Communication Overhead

General SDN architecture does not allow any control processing at SDN-switches in the core or ARs. However, this will ultimately result in excessive control overhead. It is because, as far as the mobility handling is concerned stationary nodes are also treated as MNs, where flow updates should take place despite the stability of its paths. This is due to the architectural restrictions presented in the SDN context. Currently, SDN is still in the test phase. Thus, it does not deal with large networks. However, as the size of the network expands, regular flow table updates might fail to maintain sustainable performance.

In order to tackle this problem, DMMSDN proposes the usage of abstracted IPv6 addresses to determine the stationary nodes. This mechanism is known as the *IPv6 address blocking*. IPv6 addresses of stationary nodes can be abstracted and informed to the SDN-switches in the core and the ARs, such that the flows of those nodes can be treated as general IPv6 flows. When ever a packet destined to one of such IPv6 addresses arrives at an AR, it will be forwarded without the interference of the SDN-controller. Otherwise, i.e when the first packet destined to a node having a mobility enabled IPv6 address arrives at an AR, it will be forwarded to the SDN-controller. Figure 6.39 shows an example of *IPv6 address blocking*.

6.10 Summary

This chapter suggested a novel network-based fully distributed DMM scheme called DMMSDN. DMMSDN targets an SDN environment. SDN is a paradigm compatible to achieve control/data-plane split economically. DMMSDN further utilizes DHCPv6 for accounting and IPv6 address assignment for nodes. Authentication of the MN and the relative security issues are also left for DMMSDN.

SDN suggests the logical centralization of the control-plane. However, depending on the size of the network, the control-plane might be requiring a physically distributed implementation. Up to the current data, this was not addressed in modeling the MM for SDN. Thus, this study focuses on effectively handling the MM in the SDN environment which employs a distributed set of SDN-controllers.

DMMSDN assumes that there will be several defined area network in an ISP network, each of which is handle by a single SDN-controller. Thus, DMMSDN assigns a defined block of IPv6 addresses each SDN-controller. DMMSDN suggests that out of the multiple SDN-controllers installed in the SDN environment, only one SDN-controller should keep tracks of the MN. That SDN-controller should be the one which assigns the mobility enabled IPv6 address to the MN. Address assignment, accounting, and authentication of MNs are based on the DHCPv6. When the MN moves, the other SDN-controllers detect the movement and reports to the SDN-controller which handles mobility information of the MN. Along with the mobility information, the SDN-controller also keeps records of the sessions of the MN. Thus, when a movement of the MN takes place, the SDN-controller updates all the required flows with the help of the other SDN-controllers as well. This procedure relieves the burden of the data redundancy, consistency handling, and unnecessary flow updates.

The applicability of DMMSDN and the effectiveness of multiple SDN-controllers are examined using three synthetic ISP topologies as well as mapped ISP topologies. The results confirm the compatibility of DMMSDN with respective to PMIP, DPMIP, and RO-SDN. Least redundancy of MM data, improved consistency, and faster handover can be considered as benefits of DMMSDN against PMIP, DPMIP and RO-SDN. Behavior of the SDNDMM scheme differs for ISPs based on size of the topology, different properties of the ISP, and geological distribution. Highly scattered topologies with significant amount of ARs which cover several continents benefits from highly distributed geolocation based SDN-controller installment. All the topologies show better results with multiple SDN-controller installment over single SDN-controller installment. It confirms that the communication taken place within the SDN-controller group does not reduce the performance of DMMSDN.

The event and the MN load show favorable results with higher degrees of distribution. Distribution reduces the memory occupied by the MM in each SDN-controller, while reducing the processing load of SDN-controllers as well. Thus, it can be concluded that DMMSDN is capable of handling mobility effectively in any ISP topology with an optimal number of SDN-controllers, without resulting in unnecessary controlling traffic.

Chapter 7

Conclusion

Ever growing demand for mobility in the Internet has urged Mobility Management (MM) to be performed effectively. Standard approach for MM, which introduces a centralized agent to handle the mobility, is obsoleted with the idea of Distributed MM (DMM). This is due to the fact that centralized MM suffers performance complications such as, a single point of failure and attack, scalability limitations, non-optimal routing, and unnecessary traffic. However, DMM is a comparatively new paradigm which has not reached standardization yet. The literature contains a significant number of schemes addressing DMM. As a consequence of those schemes being drafts, implementation or prototyping is yet to be performed. Therefore, simulation is identified to be the best approach to test such DMM schemes considering real world scenarios like employment in the Internet or ISP networks. Generic simulators available today, like Network Simulator 3 (NS-3) lack support to simulate non-standard DMM schemes. DMM employs novel concepts like control / data-plane split and functional distribution of control-plane. These concepts are not complemented by available generic network simulators.

This thesis introduced the design and implementation of SimNetDMM. It is a specific network layer DMM scheme simulator. As special features, it allows realistic DMM scenarios that are expected to encounter in the real world. Mobile Node (MN) movements, sessions, and MM Entities (MMEs) are drawn considering real world scenarios and administrative constraints. Thus, evaluations conducted using SimNetDMM can be considered to closely imitate the real DMM employment context.

Using SimNetDMM, performance of DMM schemes that are available as related Internet drafts in the DMM Working Group (DMM-WG) of the Internet Engineering Task Force (IETF) were analyzed. The DMM-WG can be considered as one of the main forces that works thoroughly for the standardization of DMM. The evaluation was carried out for two categories based on the scope of the mobility allowed by the selected schemes.

First, global DMM schemes were evaluated for multiple MME placement. A synthetic AS-level topology was employed during this evaluation. Then, based on the perspective of an Internet Service Provider (ISP), best secondary MME placement was identified. Employing the best and the worst MME installations and considering Mobile IPv6 (MIPv6) and MIPv6 with Return Routability (MIPv6RR) as referential schemes, Global HAHA, Migrating HA, and Distributed MIPv6(DMIPv6) were evaluated. The AS-level topology of the Internet observed by traceroute servers was employed in this evaluation. The data set was obtained from the Center for Applied Internet Data Analysis (CAIDA). Considering the results, it was concluded that the distribution in both control and data-planes known as fully distribution, is the best approach for global DMM.

Then, localized DMM schemes were evaluated using mapped ISP topologies available at Rocketfuel repository. Considering the main features of the DMM schemes, three categories are determined: fully distribution, control/data-plane separation and destination-end anchoring are identified as main categories. The results of the evaluation emphasized that the control/data-plane split yields better results in the data-plane. In contrast, it performed the worst in the control-plane, whereas the fully distribution performed best in the control-plane. However, a scheme that performed better in both the planes was not present. Thus, the vital necessity to introduce a scheme that is capable of performing better in both the planes was identified.

Based on the identified capabilities and deficiencies of the already proposed approaches, control / data-plane split based fully distribution was identified as a candidate approach for DMM. Then, due to the complications introduced by the control-plane duplicated distribution, the functional distribution of the control-plane was considered. In order to achieve control / data-plane separation, the Software Defined Networking (SDN) paradigm was considered. Then, DMMSDN was introduced targeting a distributed SDN environment where multiple SDN-controllers are installed. Each SDN-controller manages a well-defined *area network*, which hosts a per-defined set of IPv6 addresses. To assure MM information consistency and to reduce data redundancy, mobility information of a specific MN is stored only in a single SDN-controller. This SDN-controller hosts the IPv6 address assigned to the MN. The specification of DMMSDN was given and the performance of DMMSDN was confirmed using two evaluations. First evaluation compared the performance of DMMSDN against standard Proxy MIPv6 (PMIPv6), Distributed PMIPv6 (DPMIPv6), and Routing Optimization with SDN (RO-SDN). The results confirmed the applicability of DMMSDN. Then the multiple MME employment in DMMSDN was evaluated. This evaluation was conducted based on the idea that the heterogeneity in ISP networks prevail different behavior. Thus, the optimal SDN-controller installments for different ISPs were identified.

Bibliography

[1] Software-Defined Networking: The New Norm for Networks. In *ONF White Papers*. Open Networking Foundation, April 2012.

[2] 3GPP. `http://www.3gpp.org/technologies/keywords-acronyms/100-the-evolved-packet-core`.

[3] W. Aiello, F. Chung, and L. Lu. A Random Graph Model for Massive Graphs. In *Proceedings of 32nd Annual Symposium in Theory of Computing*, pages 171–180, 2000.

[4] R. Albert and Barabasi A.L. Statistical Mechanics of Complex Networks. *Reviews of Modern Physics*, pages 47–97, 2002.

[5] A. Atlas, Hares S. Halpern, J., D. Ward, and T. Nadeau. An Architecture for the Interface to the Routing System. Internet Draft, draft-ietf-i2rs-architecture-05, July 2014. Work in progress.

[6] Albert-Laszlo Barabasi and Reka Albert. Emergence of Scaling in Random Networks. *Science*, 286(5439):509–512, 1999.

[7] C.J Bernardos and J.C Zuniga. PMIPv6-based Distributed Anchoring. Internet Draft, draft-bernardos-dmm-distributed-anchoring-04, May 2014. Work in progress.

[8] J. Bi, Q. Wu, and Z. Li. Measuring the Internet Using Public Traceroute Servers. In *Proceedings of IEEE LCN*, pages 303–304, 2003.

[9] P. Bonacich. Power and Centrality: A Family of Measures. *American Journal of Sociology*, 92:1170–1182, 1987.

[10] U. Brandes. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25:163–177, 2001.

[11] BRITE Topology Generator. `http://www.cs.bu.edu/brite/download.html`.

[12] CAIDA. `http://www.caida.org/`.

[13] The CAIDA AS Relationships Dataset, 2009. `http://www.caida.org/data/active/as-relationships/`.

[14] K. Calvert, M. Doar, and E. Zegura. Modeling Internet Topology. *IEEE Transactions on Communications*, 35:160–163, June 1997.

[15] H. Chan. A Distributed Mobility Management with MobileIP. In *Proceedings of IEEE International Conference on Communications (ICC)*, pages 6850–6854, 2012.

[16] H. Chan and K. Pentikousis. Enhanced Mobility Anchoring. Internet Draft, draft-chan-dmm-enhanced-mobility-anchoring-00, July 2014. Work in progress.

[17] L. Cheng. RealNet: A Topology Generator Based on Real Internet Topology. In *Proceedings of 22nd International Conference on Advanced Information Networking and Applications - Workshops (AINAW2008)*, pages 526–532, 2008.

[18] V. Devarapalli, R. Wakikawa, and and Thubert P. Petrescu, A. Network Mobility (NEMO) Basic Support Protocol. Technical Report RFC 3963, January 2005.

[19] V. Diekert and B. Durand. Computing Minimal Multi-homogeneous Bezout Numbers is Hard. In *Proceedings of STACS 2005*, pages 244–255. Springer, 2005. LNCS 3404.

[20] E. W. Dijkstra. A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, 1:269–271, 1959.

[21] Distributed Mobility Management Working Group. `http://datatracker.ietf.org/wg/dmm/`.

[22] M. Doar. A Better Model for Generating Test Networks. In *Proceedings of IEEE GLOBECOM*, pages 86 – 93, December 1996.

[23] P. Erdos and A. Renyi. On Random Graphs I. *Publicationes Mathematicae*, pages 290–297, 1959.

[24] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On Power-law Relationships of the Internet Topology. In *Proceedings of ACM SIGCOMM*, pages 251–262, 1999.

[25] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil. Proxy Mobile IPv6. Technical Report RFC 5213, August 2008.

[26] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil. Network Configuration Protocol (NETCONF). Technical Report RFC 6241, August 2011.

[27] R. A. Hanneman and M. Riddle. Introduction to Social Network Methods. `http://www.faculty.ucr.edu/~hanneman/nettext/`, 2005.

[28] C. Hedrick. Routing Information Protocol. Technical Report RFC 1058, 1988.

[29] R. Hinden and S. Deering. IP Version 6 Addressing Architecture. Technical Report RFC 4291, 2006.

[30] Internet Engineering Task Force. `http://www.ietf.org/`.

[31] igen. `http://informatique.umons.ac.be/networks/igen/#section_download`.

[32] N. Katsutoshi, S. Isobe, T. Yagyu, and I. Akiyoshi. Implementation and Evaluation of a Network Controlled Mobility Management Protocol (IP2MM). In *Proceedings of IEEE Wireless Communications and Networking Conference*, volume 3, pages 1402 – 1408, March 2005.

[33] J. Kempf. Goals for Network-Based Localized Mobility Management (NETLMM). Technical Report RFC 4830, April 2007.

[34] J. Kempf. Problem Statement for Network-Based Localized Mobility Management. Technical Report RFC 4830, April 2007.

[35] R. Koodli. Fast Handovers for Mobile IPv6. Technical Report RFC 4068, July 2005.

[36] J. Lee and Y. Kim. PMIPv6-based Distributed Mobility Management. Internet Draft, draft-jaehwoon-dmm-pmipv6-03, Octorber 2014. Work in progress.

[37] K. Leung, G. Dommety, and Petrescu A. Narayanan, V. Network Mobility (NEMO) Extensions for Mobile IPv4. Technical Report RFC 5177, 2008.

[38] D. Liu, J.C. Zuniga, P. Seite, H. Chan, and C.J. Bernardos. Distributed Mobility Management : Current Practices and Gap Analysis. Technical Report RFC 7429, 2015.

[39] J. N. Maciel and C. D. Murta. *The Internet of the Future ; NIT: A New Internet Topology Generator*. Springer, 2009.

[40] M. Mahajan, N. Spring, D. Wetherall, and T. Anderson. Inferring Link Weights Using End-to-End Measurements. In *Proceedings of ACM IMW*, pages 231–236, January 2002.

[41] R. Mahajan, N. Spring, D. Wetherall, and T. Anderson. Inferring Link Weights using End-to-End Measurements. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement*, pages 231–236, November 2002.

[42] G. Malkin. RIP Version 2 Carrying Additional Information. Technical Report RFC 1723, 1994.

[43] J. Moy. OSPF Version 2. Technical Report RFC 2328, 1998.

[44] NetSim. `http://tetcos.com/`.

[45] M. Newman. A measure of betweenness centrality based on random walks. *Social Networks*, 27:39–54, 2005.

[46] Network Simulator : Ns3. `https://www.nsnam.org/`.

[47] NS3 DCE Mobile IPv6 Support. https://www.nsnam.org/docs/dce/manual-umip/html/index.html.

[48] OPNET. http://www.riverbed.com/products/performance-management-control/.

[49] D. Oran. OSI IS-IS Intra-domain Routing Protocol. Technical Report RFC 1142, 1990.

[50] C. Perkins. IP Mobility Support for IPv4, Revised. Technical Report RFC 5944, December 2010.

[51] C. Perkins, D. Johnson, and Arkko J. Mobility Support in IPv6. Technical Report RFC 6275, July 2011.

[52] J. Postel. Internet Protocol. Technical Report RFC 791, 1981.

[53] Route Information Service. https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris.

[54] RocketFuel. http://research.cs.washington.edu/networking/rocketfuel/.

[55] RocketFuel Interactive Maps. http://research.cs.washington.edu/networking/rocketfuel/interactive/.

[56] Public Route Server. http://routeserver.org/.

[57] RouteViews. http://www.routeviews.org/.

[58] D. Savage, C. Slice, J. Ng, S. Moore, and R. White. Enhanced Interior Gateway Routing Protocol. Internet Draft, draft-savage-eigrp-02, April 2014. Work in progress.

[59] P. Seite, H. Yokota, J. Korhonen, H. Chan, and D. Liu. Requirements for Distributed Mobility Management. Technical Report RFC 7333, 2014.

[60] M. Shand and L. Ginsberg. Reclassification of RFC 1142 to Historic. Technical Report RFC 7142, 2014.

[61] H. Soliman, C. Castelluccia, and and Bellier L. ElMalki, K. Hierarchical Mobile IPv6 (HMIPv6) Mobility Management. Technical Report RFC 5380, October 2008.

[62] N. Spring, M. Mahajan, and T. Anderson. Quantifying the Causes of Path Inflation. In *Proceedings of ACM SIGCOMM*, pages 113–124, 2003.

[63] N. Spring, M. Mahajan, and D. Wetherall. Measuring ISP Topologies with Rocketfuel. In *Proceedings of ACM SIGCOMM*, pages 2–16, August 2002.

[64] N. Spring, R. Mahajan, and T. Anderson. Quantifying the Causes of Path Inflation. In *Proceedings of ACM SIGCOMM*, pages 113–124, 2003.

[65] Traceroute. `http://www.traceroute.org/`.

[66] J.P. Vasseur, N. Shen, and Aggarwal R. Intermediate System to Intermediate System (IS-IS) Extensions for Advertising Router Information. Technical Report RFC 4971, July 2008.

[67] R. Wakikawa and G. Valadon. Migrating Home Agents Towards Internet-scale Mobility Deployments. In *Proceedings of the ACM 2nd Conference on Future Networking Technologies (CoNEXT)*, 2006.

[68] R. Wakikawa, Z. Zhu, and L. Zhang. Global HA to HA Protocol Specification. Working Draft, September 2011. Work in progress.

[69] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*. New York, Cambridge University Press, 1994.

[70] B.M. Waxman. Routing of multipoint connections. *IEEE Journal on Selected Areas Communication*, 6:1617–1622, 1988.

[71] J. Winick and S. Jamin. *Documentation of Inet- 3.0: Internet Topology Generator*.

[72] H. Yang and Y. Kim. Routing Optimization with SDN. Internet Draft, draft-yang-dmm-sdn-dmm-01, April 2014. Work in progress.

[73] A. Yegin, K. Kweon, J. Lee, and J. Park. Corresponding Network Homing. Internet Draft, draft-yegin-dmm-cnet-homing-02, July 2014. Work in progress.

[74] Z. Zhu, R. Wakikawa, and L. Zhang. Survey on Mobility Support in Internet. Technical Report RFC 6301, July 2011.