TRAFFIC ENGINEERING FOR LISP-ENABLED NETWORKS

A THESIS
IN
Computer Science

Presented to the Faculty of the University
of Missouri–Kansas City in partial fulfillment of
the requirements for the degree

MASTER OF SCIENCE

by
RAGHUNANDAN SRIDHAR
M. S., University of Missouri – Kansas City, 2013

Kansas City, Missouri
2013

TRAFFIC ENGINEERING FOR LISP-ENABLED NETWORKS

Raghunandan Sridhar, Candidate for the Master of Science Degree

University of Missouri–Kansas City, 2013

ABSTRACT

Inter-Domain Traffic engineering in the Internet faces serious limitations because of the current IP routing and addressing architecture. This coupled with Border Gateway Protocol's (BGP's) way of selecting performance-blind paths forces ISPs to de-aggregate IP prefixes to control the flow of packets between ASes. Advertising such de-aggregated, surplus prefixes for local benefits is causing the routing table of the Default Free Zone (DFZ) to grow rapidly, which contribute to routing scalability problems. Recently, in order to address this scalability issue, LISP (Locator/Identifier Separation Protocol) has been proposed, which separates an address space into a non-routable *End-Point Identifiers* (EIDs) and a routable *Routing Locators* (RLOCs), where each EIDs can be associated to more than one (multiple) RLOCs. In this work, we discuss two optimization models for traffic engineering in LISP-enabled network which exploits the route diversity or the path diversity the LISP inherently provides by introducing the concept of *grouping* multiple RLOCs with traffic proportioning or load-balancing as the optimization criterion. We compare the models to the base case that identifies with the current routing architecture (i.e. no proportioning). Through our study, we observe that LISP-based traffic engineering with multiple RLOCs offers noticeable benefits

compared to when we do traffic engineering without proportioning demands to multiple RLOCs, except when the network is uniform in terms of load and capacity.

APPROVAL PAGE

The faculty listed below, appointed by the Dean of the School of Computing and Engineering, have examined a thesis titled " Traffic Engineering for LISP-enabled Networks" presented by Raghunandan Sridhar, candidate for the Master of Science degree, and hereby certify that in their opinion it is worthy of acceptance.

Supervisory Committee

Deep Medhi, Ph.D., Committee Chair
Department of Computer Science & Electrical Engineering

Cory Beard, Ph.D.
Department of Computer Science & Electrical Engineering

Baek-Young Choi, Ph.D.
Department of Computer Science & Electrical Engineering

CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

ACKNOWLEDGEMENTS

It is with immense gratitude that I thank and acknowledge my academic advisor Dr. Deep Medhi for his support and guidance during my thesis research work. This thesis is my first research, and Dr.Medhi has given me invaluable mentoring starting from finding the research topic to writing papers. On a personal note, it was an honor and a privilege for me to learn from Dr. Medhi and I believe that by working under him I have acquired a solid foundation to excel in the field of Computer Networking. On the other hand, I would like to thank and extend my deepest appreciation to Dr. Baek-Young Choi and Dr. Cory Beard from the Department of Computer Science and Electrical Engineering for their advice on my thesis work.

I would also like to thank my lab mates at Networking Research Laboratory (NetRel) at School of Computing and Engineering for their inputs and critiques. Specifically, I am thankful to my colleague Soumick Dasgupta for his valuable insights and suggestions which has helped me to successfully design and implement my thesis work.

Furthermore, I am indebted to my Parents, my Wife and my Friends for their unconditional support and encouragement throughout my journey towards completing my Master's program.

CHAPTER 1

INTRODUCTION

Since from the early development of the Internet much of the effort has been dedicated to addressing the issues related to IP numbering space i.e. size of the IPv4 address, which is not unusual given the rapid development of the Internet. Technologies like Private Addressing and NAT (Network Address Translation) have come in handy to appease the rate of exhaustion of IPv4 addresses and with the birth of IPv6 (Internet Protocol Version 6) IETF had made sure that the world will never ran out of IP addresses. However, because of one the major loophole in the current Internet routing architecture, when I say loophole, it is with respect to IP address semantics where an IP address is used as both “End point identifier” and as well as “Routing locator”, today’s Internet routing and addressing system is facing serious scaling problems, i.e. the routing table size of the Default Free Zone (top tier ISP’s routing table, to be precise) is growing at an alarmingly rapid rate.

To discuss this scalability problem, IAB (Internet Architecture Board) held a “Routing and Addressing” workshop in October of 2006 and listed out multiple factors which are directly influencing the rapid growth of DFZ routing table. Below are those following factors.

Factors influencing the rapid growth of DFZ routing table size:

- Multi-homing
- Traffic Engineering
- Non-aggregatable address allocations

- Business Mergers and Acquisitions

Of these above factors, it has been measured that Multihoming and Traffic Engineering are two major contributors towards the rapid growth of DFZ routing table as they lead to prefix de-aggregation and/or the injection of unaggregatable prefixes into the DFZ RIB [1].

The below graph shows size of the Forward Information Base (FIB) and the term "super-linear" has been used to characterize its growth. It is estimated that the size of the current BGP routing table (RIB) is over 600,000 entries and that of the FIB is a little over 300,000 with both increasing exponentially [1]. Thus, the super-linear growth in the routing load presents a scalability challenge for current and /or future routers.

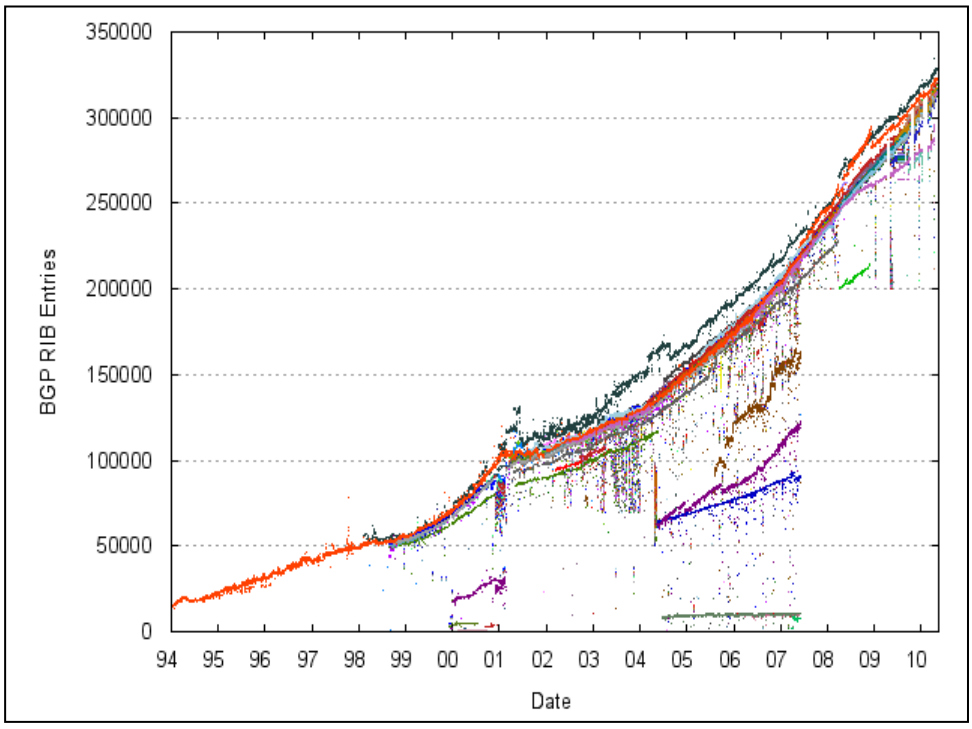

Figure 1: Super- linear growth of BGP FIB

In the next section, we briefly describe all the 4 factors and in chapter 3 "Current Traffic Engineering Practices" we look in detail at how today's traffic engineering practices are polluting the Internet by advertising more specific prefixes for local benefits at a global cost.

1.1 Multihoming

Multihoming, as the name suggests is a case where a single site is served by more than one Internet Service Provider (ISP). Multihoming offers many advantages compared to single-homed sites, of which, most notable once are load balancing and back up routing, which addresses single point failure.

Multihoming can be achieved using either Provider Independent (PI) address numbering or by Provider Assigned (PA) Address numbering and Table 1 gives the difference between these two:

Table 1: Providers Independent v/s Providers Assigned Address Space

Providers Independent (PI) Address Numbering	Providers Assigned (PA) Address Numbering
With PI, the end site directly request the RIR for a chunk of addresses independent of its provider address space, thereby avoiding the scenario of renumbering its devices when they wish to change its existing ISP.	With PA, the end site obtains its chunk of addresses from the Providers address space and hence has to renumber all of its end devices when they wish to change its existing ISP.
With PI, the ISP's have to advertise all of end site prefixes as the address numbering is independent of the ISP's address space.	With PA, the ISP's will only need to advertise the summary of all of end site prefixes as the addresses are aggregatable.

Even though with PA address space where end site prefixes are assigned and only these aggregated addresses are propagated into the DFZ, we discuss below that the choice of PI v/s PA space has no impact on the control plane load.

The current Internet routing uses a blunt instrument called "longest matching prefix/routing" [9], where data will be routed through those links which advertises the

more specific prefix than the one which advertises the less specific one. Below Fig. 2 shows the effect of this kind of routing, here we have 2 AS's namely; AS1 and AS2 both are multihomed with Provider A and Provider B providing services to AS1 and Provider C and Provider D providing services to AS2, please note that AS1 uses PA address space and AS2 uses PI address space.

For AS1, provider A is a primary ISP and provider B is used as a back-up ISP, hence to route traffic to 20.1.1.0/24 prefix provider A only advertises the aggregate 20.0.0.0/8, thus all the traffic addressed to 20.1.1.0/24 prefix will come through provider A. Now, AS1 wants to load balance the traffic coming into it through both provider A and provider B and hence request provider B to advertise 20.1.1.0/24 prefix (provider B has to advertise specific prefix /24 because it is not aggregatable based on the address space it is using). Now, because of "longest match routing" as explained earlier now all the traffic addressed to 20.1.1.0/24 will come through provider B but to achieve the load balancing criterion provider A have to advertise additional more specific prefix 20.1.1.0/24 along with 20.0.0.0/8 thereby aiding local benefit at a global cost and hence polluting the Internet.

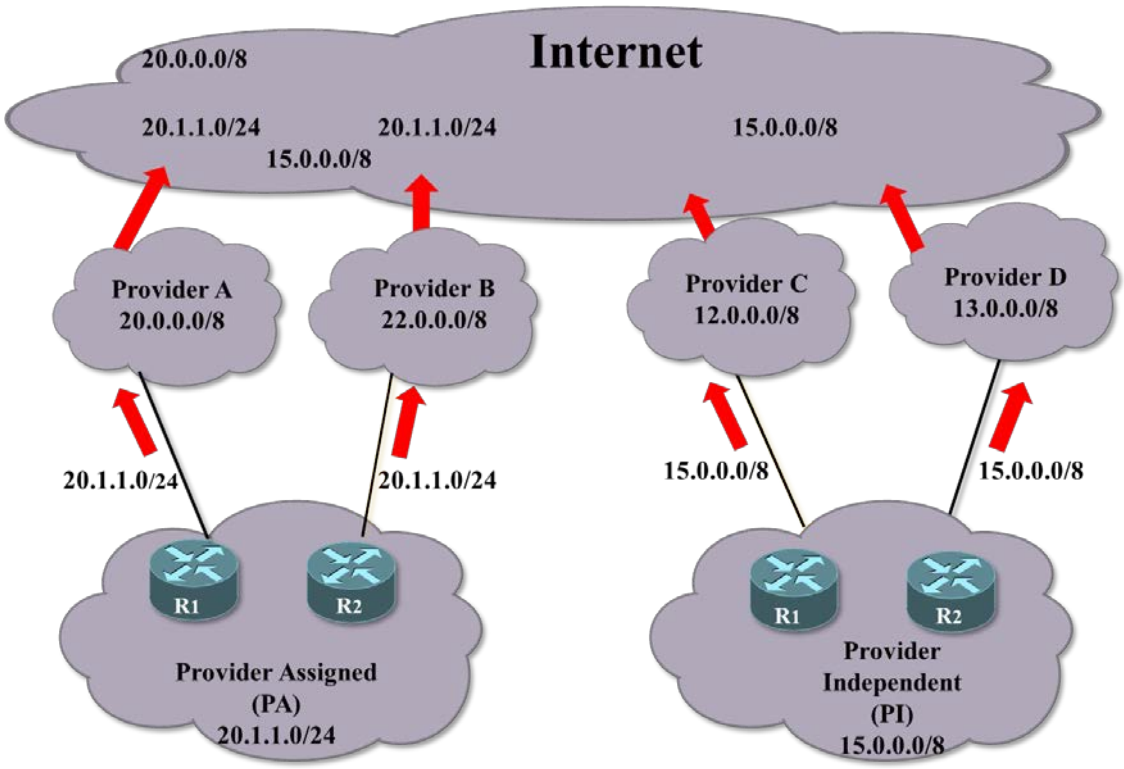

Figure 2: Polluting the Internet through Multihoming

1.2 Traffic Engineering

Traffic Engineering (TE) is an act of arranging for certain Internet traffic to pass through or avoid certain network paths, where, the selection of these paths are influenced by a set of performance objectives to achieve better user performance and efficient use of network resources [9]. One such performance objective is “Load Balancing”, where ISP’s spread their traffic load across multiple paths subject to available link capacities.

We differentiate Traffic Engineering into two types based on whether the traffic is entering or leaving an AS:

1.2.1 Outbound Traffic Engineering

Outbound TE as the name suggest involves controlling the traffic leaving an AS either by tweaking the metrics of internal Interior Gateway Protocol (IGP) to choose the shortest exit for two equally good BGP paths or by using one of the attributes of BGP called LOCAL_PREF, which is a metric used internally within an AS between BGP speakers, where this attribute helps in identifying a specific outgoing BGP speaker when a AS has connectivity to multiple ASes or multiple BGP routes even with the same next hop AS [5]. An example of usage of BGP path-attribute LOCAL-PREF to achieve Outbound TE is as shown below.

In the Fig. 3 below, IP prefix 20.20.0.0/16 originated from AS1 is advertised to ASes AS2 and AS3 which in turn advertises the IP prefix to AS4, which arrives at BGP speakers R1 and R2, respectively. Now, if AS4 wishes to channel the traffic destined to IP prefix 20.20.0.0/16 only through R2, it can do so by introducing local preference, where BGP speakers R1 and R2 are configured with LOCAL-PREF values which are internally communicated to IBGP speaker R3, thus, when a user traffic arrives at R3 destined to IP prefix 20.20.0.0/16, it will prefer to use the outgoing BGP speaker R2 since the LOCAL-PREF metric value is higher for this router compared to R2.

Thus, because Outbound TE is achieved using a site's own IGP or/and using BGP path attribute like LOCAL-PREF it does not impact routing outside of a site and hence do not influence the growth of the routing table in DFZ.

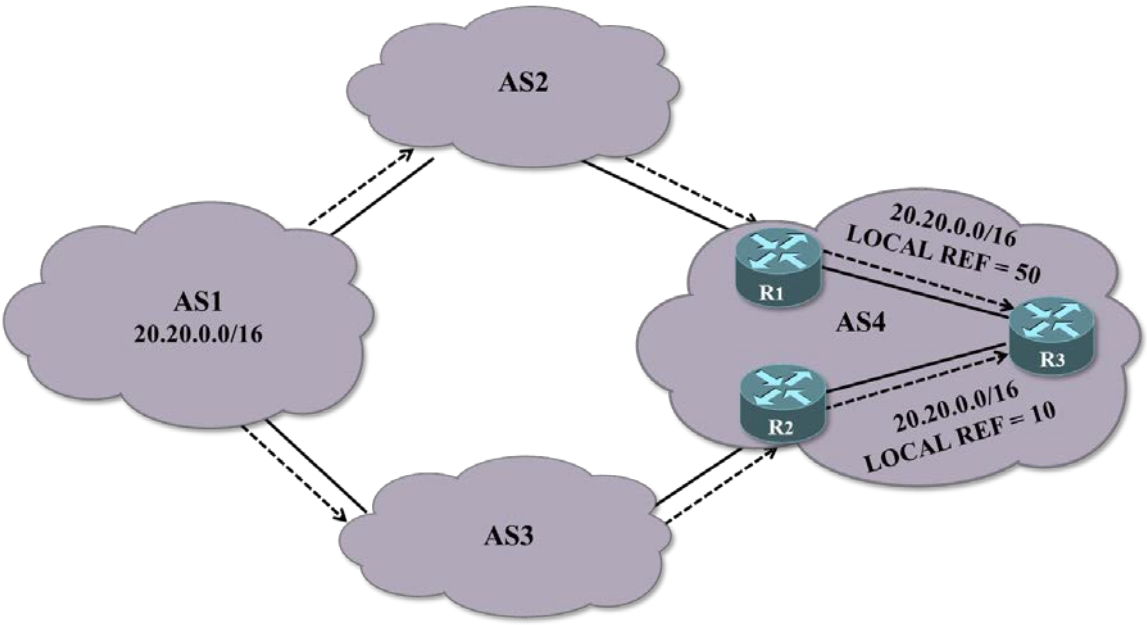

Figure 3: Outbound Traffic Engineering using BGPs LOCAL-PREF Attribute

1.2.2 Inbound Traffic Engineering

Inbound TE generally refers to Inter-domain Traffic Engineering where traffic engineering is achieved by announcing a more-specific route along the preferred path that captures the desired traffic and channels it away from the regular/defined path it would take otherwise [9]. In today's Internet majority of Inbound TE is achieved by using BGP attributes such as AS-PATH prepending and Redistribution Communities. In chapter 3 "Current Traffic-Engineering practices" we will discuss in detail the practices and drawbacks of Inter-domain traffic engineering.

1.3 Non-Aggregatable Address Allocations

Site's which wish to have more than one provider (Multihoming) to satisfy mission-critical business applications would like to use Provider Independent address space as this would remove the burden of re-assigning addresses to end points when they wish to change the provider [9]. To achieve this, the multihomed site's request the RIR for a chunk of IP addresses which is independent of their respective provider's address space (PI) thus forcing their respective provider's to advertise a large number of specific-prefixes as they cannot aggregate these prefixes and thereby polluting the Internet.

1.4 Business Mergers and Acquisitions

When acquisitions and merges takes place for business reasons, a Company that buy out or merges with other organizations may soon find out that its network assets are numbered out of many different and un-aggregatable address blocks [9]; thus they no longer be able to advertise a single aggregate there by forced to advertise more specific prefixes and there by indirectly contributing towards the growth of DFZ's routing table.

1.5 Approaches Considered

Over the years, considerable effort was put in to identifying solutions for the scalable inter-domain routing for the Internet, some proved to be dead end and other triggered new ideas, here in this section we investigate these approaches and evaluate their pros and cons.

1.5.1 MULTI6

The MULTI6 working group explored the solution space for scalable support of IPV6 multihoming. Their solutions revolved around two ideas: the allocation of providers-independent (PI) address space for customers and the second, assigning multiple address prefixes to multihomed sites i.e. use of both ISP address spaces and when one fails the communication is moved to the other address providing protection against single point failure. The solutions proposed were technically flawed as the first solution was not scalable because with PI address space ISP's cannot advertise a single aggregate, the second introduces fundamental changes to the Internet routing system, and thus, this approach was deemed incomplete [9].

1.5.2 SHIM6

The SHIM6 working group took the second approach from MULTI6, i.e. supporting multihoming through the use of multiple addresses and introduced host-based approach, where the host IP stack includes a "shim" which provides a stable "upper-layer identifier" (ULID) to the upper layer protocols above IP and may involve rewriting IP packets sent and received to facilitate currently working IP address to be used in the transported packets, i.e. it changed the current design where a single IP address was used as both locator and an identifier by the end systems, with SHIM6, the upper layer protocols above IP used 128-bit ULID called shim to identify endpoints (e.g. TCP connections)and the 128-bit IPv6 address was used as a locator [9]. With this locator and identifier separation SHIM6 isolated the upper layer protocols from multiple

IP layer addresses thus enabling multihomed sites to use provider-aggregatable address space thus facilitating provider-based prefix aggregation.

Even though SHIM6 addressed scalability issues it had many drawbacks. First, with the introduction of “shim” all host stack implementations requires modifications to support shim processing. Second, less support for traffic engineering at ISP level as the SHIM6 is a host-based approach. Third, the identifier (ULID) and locator approach mandates host to keep track of state information regarding multiple locators of the remote communication end, which is fine with respect to individual hosts but will introduce significant problems on large application severs which handles thousands of simultaneous TCP connections. Finally, as SHIM6 solution encourages multi-homed site to use provider-allocated address space this also introduces major issues when a site wish to change its provider as they will be forced to renumber their end points based on the providers address space [9].

In our next section we will study in detail one of the newest and the most accepted approaches called “Locator/Identifier Separation Protocol” (LISP) and see how this approach tackle the scalability issues and provides better solution for traffic engineering and multihoming problems.

CHAPTER 2

LOCATOR/IDENTIFIER SEPERATION PROTOCOL (LISP)

LISP is a simple IP-to-IP tunneling protocol which aims to solve the routing scalability issue by splitting the current single IP address space into 2 new numbers: Routing Locators (RLOC's) and Endpoint Identifiers (EID's) [2]. Both RLOCs and EIDs are syntactically-identical to the IP addresses but the semantics of how they operate are different. To support this locator/identifier split LISP defines functions for mapping between the two numbering spaces when a packet travels from source to destination. One of the most important features of LISP is it is incrementally deployable and do not need any changes to the current host protocol stack or to the core of the Internet infrastructure. Before we explore the working of LISP, we need to understand few LISP terminologies so the next section called "LISP Terminologies" is dedicated for this purpose.

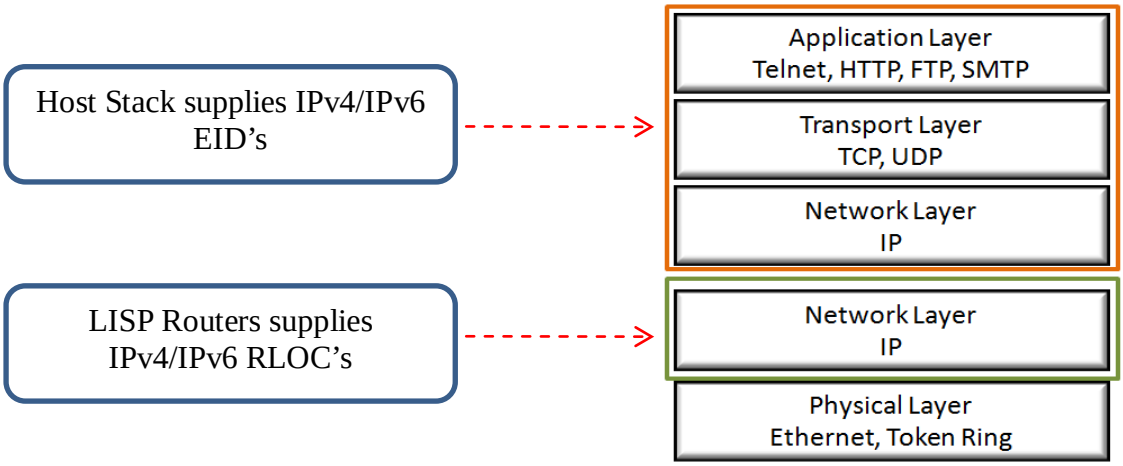

Figure 4: LISP Architecture

2.1 LISP Terminologies

- Routing Locator (RLOC): An RLOC is an IPv4 or IPv6 address of an Egress Tunnel Router (ETR) inside a LISP site. RLOC is like public addresses which are advertised into the DFZ and are aggregatable based on the PA address space. Inside a LISP site multiple RLOC's can be assigned to the same ETR device or to a multiple ETR devices [2].
- Endpoint Identifiers (EIDs): An EID is 32-bit IPv4 or 128-bit IPv6 value residing in the inner most LISP header of a packet. EIDs are independent of providers address space and may have site-local address structure to facilitate desired routing within the site and hence are not visible to the global routing system just like todays private addresses. An end system inside a LISP site uses DHCP to obtain a source EID and does a DNS lookup to obtain a desired destination EID. Note that a single EID may be associated to multiple RLOCs inside the same LISP site [2].

- Ingress Tunnel Router (ITR): An ITR as the name suggest is a border router at the LISP site which handles all the outgoing packets. Once an ITR receives a packet from one of the end system, it looks in to the destination filed of the IP packet and obtains the destination EID; with this information the ITR does a mapping lookup on to obtain its corresponding destination RLOC, if the map look-up is successful [2].

  ITR encapsulates the IP header with the LISP header and fills the destination address field of the LISP header with the obtained destination RLOC address and puts its own address in the source field, note that this destination RLOC may be an intermediate routers address called proxy router which may a better knowledge of the EID-to-RLOC mapping of the destination RLOC.
- Egress Tunnel Router (ETR): Contrary to an ITR, ETR receives the incoming LISP-encapsulated packet (only if it is addresses to one of its RLOCs) and strips the “outer” LISP header and finally forwards the packet to one of the end systems based on the IP address of the “inner” IP packet [2].
- EID-to-RLOC Database: This is a global distributed database which contains the all known EID-Prefix to RLOC mappings [2].
- EID-to-RLOC Cache: This is a small, dynamic and short-lived table that an ITR stores. Each ITR is responsible for tracking, timing-out and validating EID-to-RLOC mappings that thy store in their respective cache [2].
- LISP Header: LISP header is comprised of an IPv4 or IPv6 header, an UDP header and a LISP-specific 8-byte header following the UDP header. An ITR encapsulates an IP packet with this LISP header while the ETR strips it [2].

- Negative Mapping Entry: This message or code is returned either by an ITR or an ETR when there is no EID-to-RLOC mapping in their respective cache or database. Specifically, this type of entry is used to describe a prefix from a non-LISP site, which is absent from the mapping database [2].
- Proxy ITR (PITR): PITR is used for “Interworking” between a non-LISP and LISP site, where, PITR acts as a proxy for a non-LISP site and encapsulates the IP packet with the LISP header and forwards it to an appropriate ETR [2].
- Proxy ETR (PETR): Similar to PITR, PETR acts a proxy for non-LISP site by striping the outer LISP header and forwards the packet to the appropriate IP address [2].
- LISP Site: Is a set of routers under single administration and acting as a demarcation points to separate the edge network from the core network [2].

2.2 LISP Header Structure

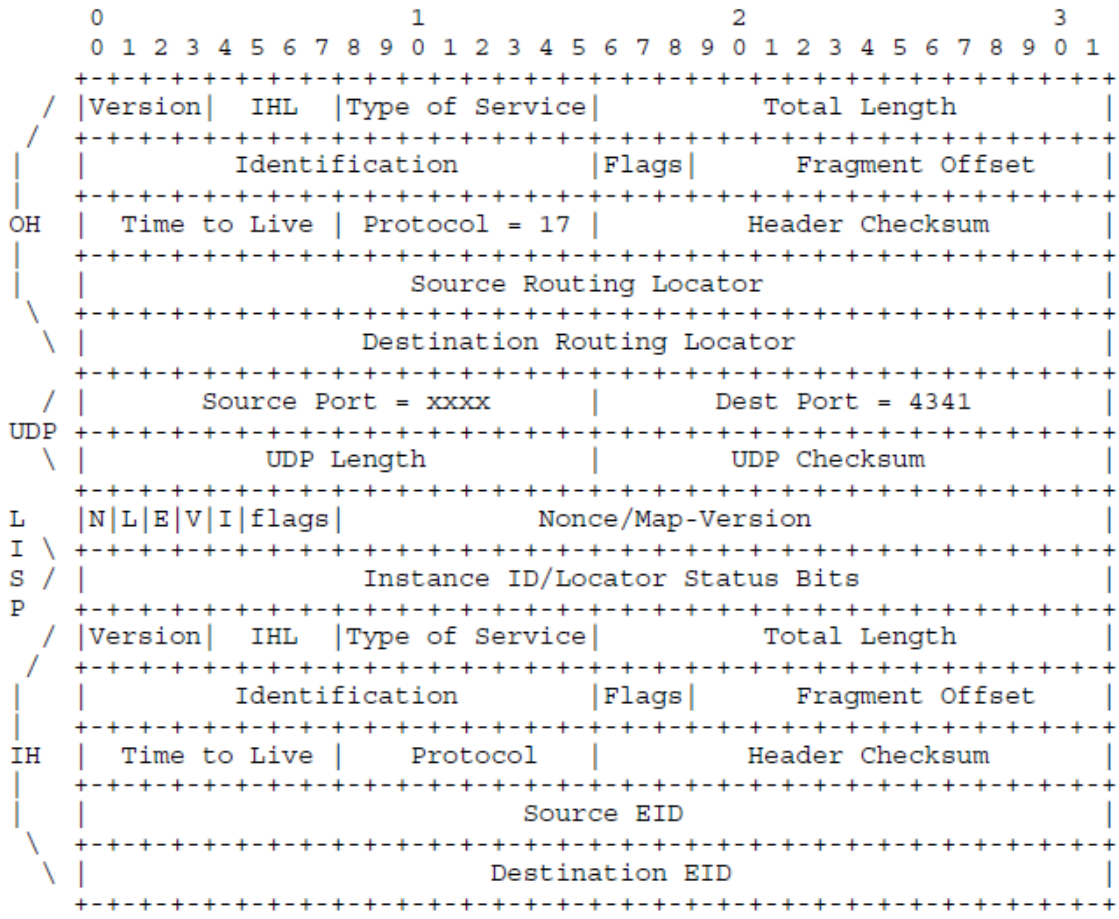

Figure 5: LISP IPv4-in-IPv4 Header Format

Figure 6: LISP IPv6-in-IPv6 Header Format

The Fig. 5; above shows the LISP IPv4-in-IPv4 header format which is comprised of an outer IPv4 header plus a UDP header plus a LISP header plus the inner IPv4 header [2]. In this section, we will look at the important difference between the

outer and the inner IPv4 headers and will provide a detail description of the fields related to UDP and LISP header; especially we will look into the flag fields of LISP header and the role that they play in routing the LISP packets between two end systems.

Table 2: Inner IPv4 Header v/s Outer IPv4 Header in a LISP Encapsulation

Inner IPv4 Header	Outer IPv4 Header
Represents the header on the datagram received from the originating hosts, where the source and the destination IP address are source and destination EIDs.	Represents the header prepended by an ITR, where the source and the destination IP addresses are one of the RLOCs of the respective ITR and ETR.
It is either 32-bit or 128-bit depending on IPv4 or IPv6 packet.	It is either 32-bit or 128-bit depending on what the respective ITR or ETR supports.
The IP protocol number depends on the type of Layer-3 protocol that is being used to communicate.	The IP protocol number is always 17, which identifies UDP.
The value of the DF bit is implementation specific with respect to IPv4 packets.	The value of the DF bit is set to 0 or 1 depending on whether a "Stateless Solution" or a "Stateful Solution" is defined for MTU handling.

2.2.1 UDP Header Field Description in a LISP Encapsulation

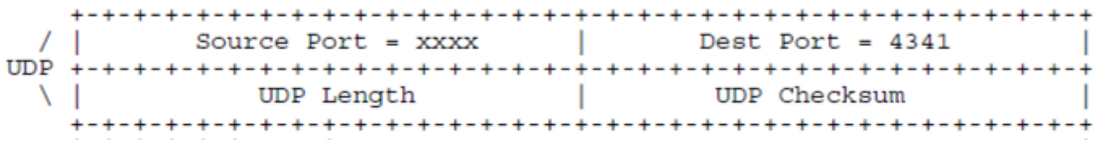

Figure 7: UDP Header Fields in LISP IP-to-IP Header Format

- Source Port: The value of the source port is determined by an ITR by applying a hash algorithm on 5 tuple [2] as below

Source Port = Hash_algorithm (Source address, destination address, source port, destination port, IP protocol number field)

Where: The 5 tuple values are from the inner IPv4/Ipv6 header.

- UDP Checksum: Table 3, describes the values that an ITR puts into this field and defines the action taken by an ETR based on the written values.

Table 3: Action taken by an ETR against the value of the UDP Checksum

UDP Checksum Values	Action Taken by an ETR
Zero	When an ETR receives a packet with a UDP checksum value of zero, the ETR MUST accept the packet for decapsulation.
Non-Zero	When an ETR receives a packet with a non-zero UDP checksum, it may verify the checksum value and if the verifications fails the packet will be dropped silently else it process with the decapsulation.

- UDP Length: The value differs based on whether the encapsulated packet is an IPv4 or IPv6 packet.

  IPv4 Encapsulated Packet: IPv4 encapsulated Packet + inner + header total length + UDP Header + LISP Header.

  IPv6 Encapsulated Packet: Inner header payload length + IPv6 encapsulated Packet + UDP Header + LISP Header.

2.2.2 LISP Header Field Description in a LISP Encapsulation

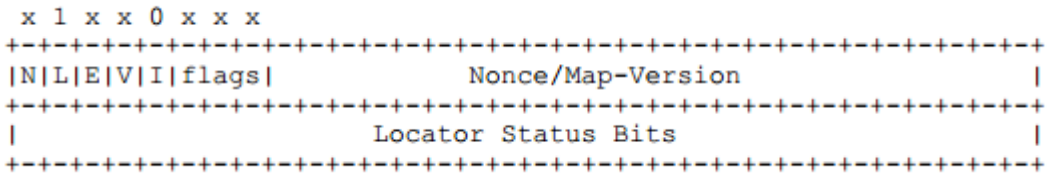

Figure 8: LISP Header Fields

Below Table 4: Usage of flags in LISP Header filed provides a detail description on each flags, their dependencies and their purpose.

Table 4: Usage of Flags in LISP Header Filed

Flag	Dependency Flag/s	Purpose
N	E	This flag is mainly used to provide or implement a way to identify connectivity between an ITR and the corresponding ETR when data flow is bi-directional. Implementation: ITR: When set along with 'E' flag ITR includes a 24-bit Nonce" value requesting for nonce echo. ETR: ETR responds to nonce echo request from an ITR, with a data its next data packet with flag values of N = 1, E = 0.
L	None	When set (L = 1) indicates the presence of "Locator-

Flag	Dependency Flag/s	Purpose
		Status-bits” in the header.
E	N When, N = 1, E must be 1. When, N = 0, E bit should be ignored.	ITR: Sets to one (E = 1) to request for “Echo Nonce” from the corresponding ETR. ETR: Sets to zero (E = 0), when echoing nonce value back to the corresponding ITR.
V	N When, V – 1, N must be 0.	Used for the purpose of MAP-version validation between an ITR and its corresponding ETR.
I	None	This instance bit is used to provide protection against usage of “Private Address” by more than one organization inside a single LISP site. ITR: Sets to 1 and places a 24-bit LISP router value which uniquely identifies the address space. Note: When set (I = 1) , the locator status bit is reduced from 32 bits to 8-bits and the higher order 24-bits is used as an Instan ce ID. ETR: When set (L = 1) uses the instance ID value to identify the correct forwarding table to use for the inner destination EID lookup.

- LISP Nonce

This field value is randomly generated by an ITR to verify ITR-to-ETR reachability.

- LISP Locator Status Bits

This field value is set by an ITR to indicate an ETR the up/down status of the Locators on the source site. The Locators Status bits are numbered from 0 to n-1 from the least significant bit of field, where a status of 1 indicating the RLOC associated with that bit ordinal has up status. When I bit is set this filed value is reduced from 32-bits to 8-bit [2].

2.3 LISP Control Plane Packet Formats

The below figures (Fig. 9 and Fig.10) shows the LISP Control Plane Packet format for both IPv4 and IPv6 packets respectively, where each is used to retrieve the EID-to-RLOC Mapping information.

The UDP header fields have the following values for source and destination ports depending on whether the packet is a Map-Request packet or a Map-Reply packet.

Map-Request Packet

Source Port – Arbitrarily chosen by the sender (ITR).

Destination Port – Destination port value would be 4342.

Map-Reply Packet

Source Port – Source port value would be 4342

Destination Port – Arbitrarily chosen by the sender (ETR).

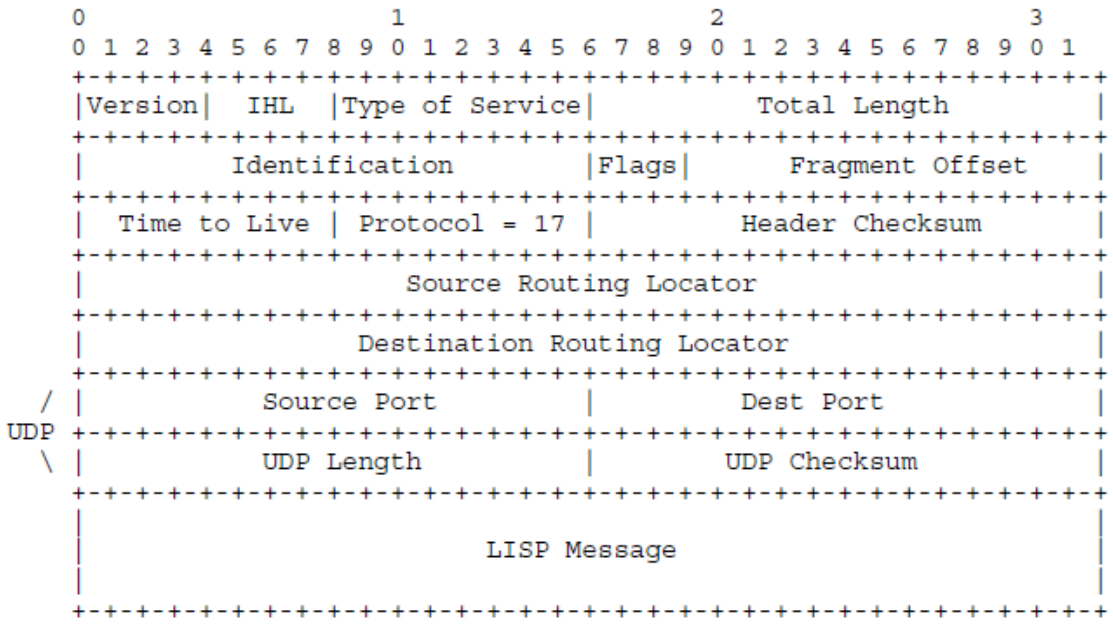

Figure 9: LISP IPv4 Control Plane Packet

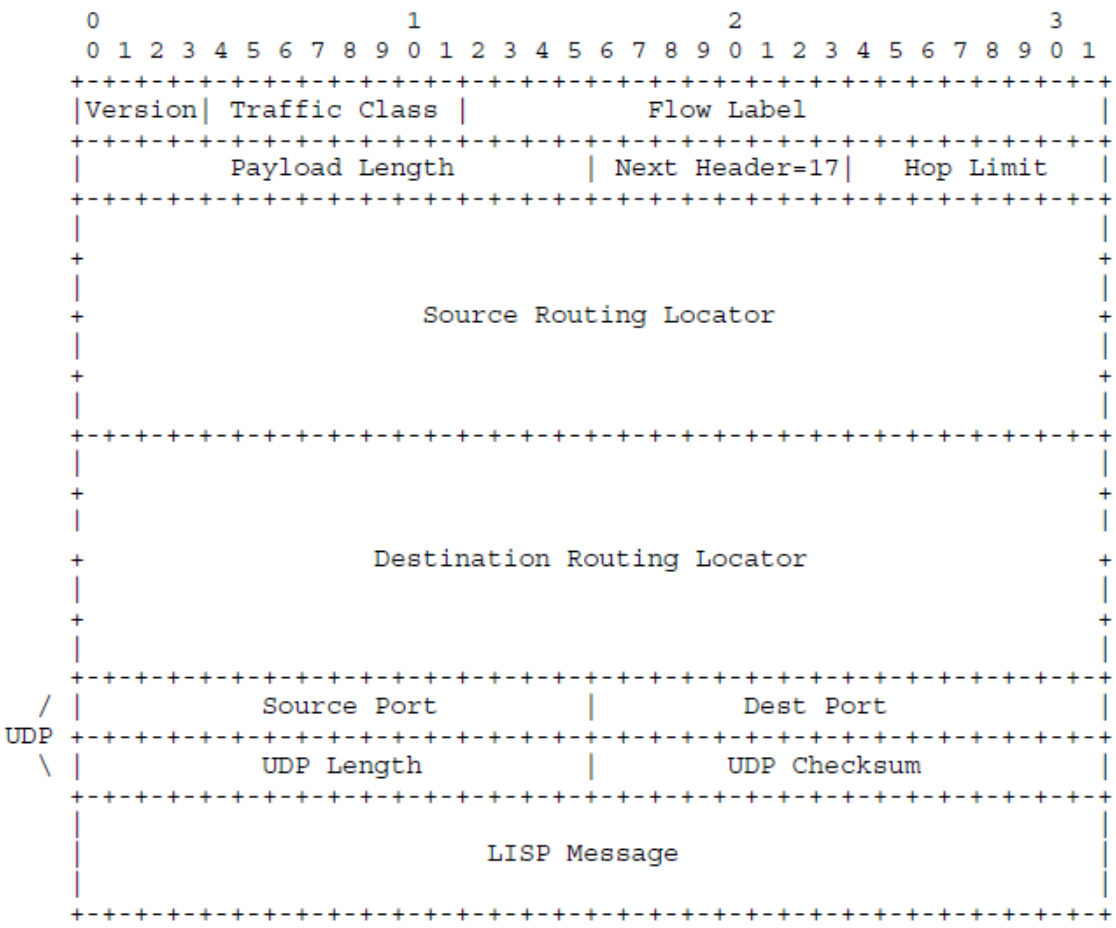

Figure 10: LISP IPv6 Control Plane Packet

2.3.1 LISP Map-Request Packet

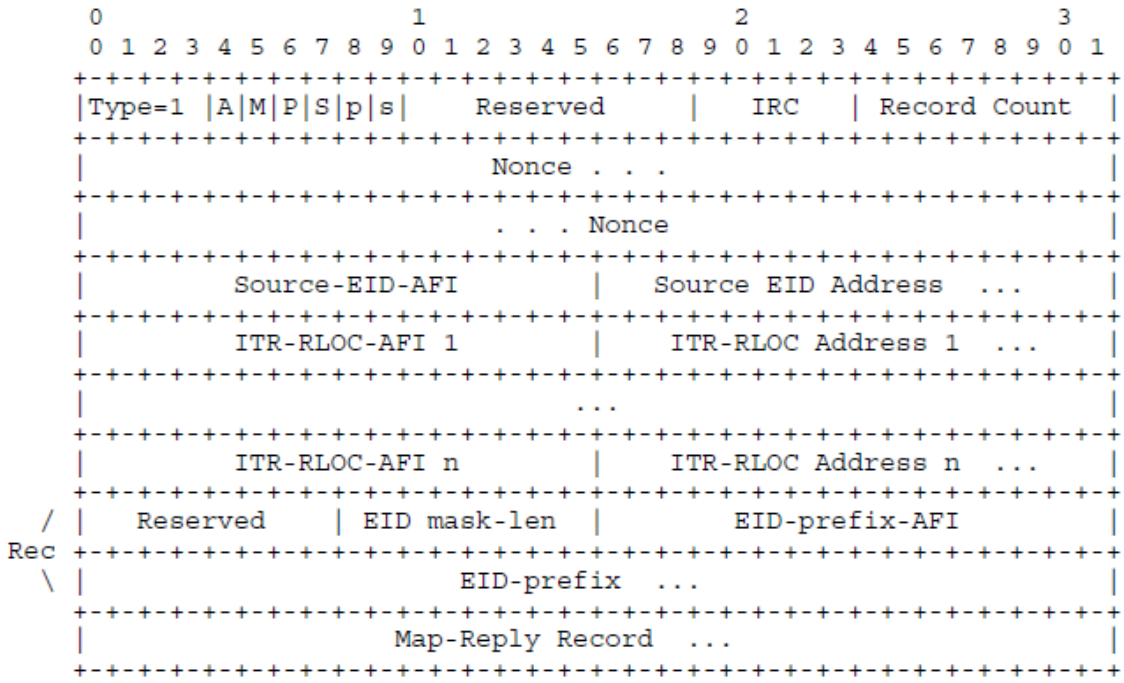

Figure 11: LISP Map-Request Message Format

Table 5 shown below describes the flag values associated with the LISP Map-Request Packet.

Table 5: Usage of Flags in LISP Map-Request Packet

Flag	Purpose
A	Authoritative bit, always set to zero (A =1) for UDP-based Map-Request propagated by an ITR.
M	Indicates the presence of Map-Reply record segment, when set (M =1).
P	Probe-bit, when set (P=1) the respective ETR should treat this Map-Request packet as a Locator reachability probe and should reply with probe-bit set (P = 1), indicating Map-Reply packet is a Locator reachability probe reply.
S	Solicit-Map-Request bit, used by an ETR to advertise the changes in the EID-to-RLOC mappings at their respective site to a recently communicated ITR. This type "push" model is used by an ETR to control the rate at which they receive the of Map-Request messages.

- Nonce

  This field value plays a very important role with respect to the security of the LISP mapping protocol, created by the sender of the Map-Request and it is generally 8-byte long [2].

2.3.1.1 EID-to-RLOC UDP Map-Request Message

The Map-Request messages can be of 3 fold as below:

1. When an ITR does not have a mapping entry for the requested destination EID.
2. When an ITR wants to test for RLOC reachability.
3. When an ITR wants to clear out the stale entries in the caching i.e. refreshing the mapping entries before the TTL expires.

2.3.2 LISP Map-Reply Packet

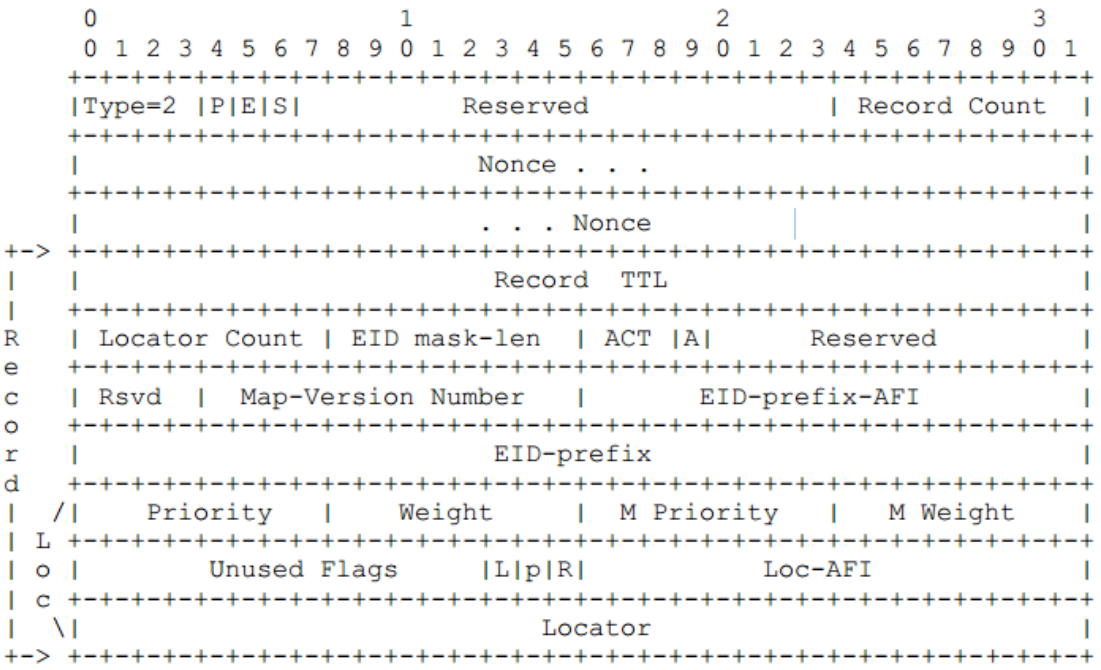

Figure 12: LISP Map-Reply Message Format

The most important part of the Map-Reply Message is the one labeled as Record, which includes critical information regarding EID-RLOC mapping for a respective Map-Request Message, below we will describe the important fields and their meaning within the Record.

- Record TTL

This field represents the amount of time in minutes an ITR receiving this Map-Reply message should cache the appropriate entry, where a TTL value of “zero” [2] would cause an ITR to remove the entry from the cache immediately and a TTL value of “0xffffffff” [2], allows an ITR to decide the locally how long to store the mapping.

- ACT

  When the “Locator Count” field is set zero, this 3-bit field specifies “Negative Map-Reply Actions” [2] to an ITR. The current active values are as below

  - (0) No Action: If a map cache entry is present an ITR ignores the TTL value of the entry and encapsulates based on the mapping in the cache.
  - (1) Natively Forward: The packet from the source EID is neither encapsulated with an LISP header nor dropped but natively forwarded.
  - (2) Send-Map-request: Prompts for a Map-Request packet from the ITR.
  - (3) Drop: The packet from the source EID matching this cache entry is dropped.

- Priority

  Each RLOCs associated with an EID has a priority value, where higher values are less preferable. This field plays an important role with respect to Traffic Engineering where RLOCs with same priority can be used to “load-balance” [2] the traffic between the RLOCs.

- Weight

  This field defines how to load-balance the traffic among different RLOCs with same priority values.

2.4 LISP+ALT Architecture

LISP Alternate Logical Topology (ALT) is a “Mapping Service Interface” used to find the appropriate mapping information between an End-Point Identifier (EID) and a Routing-Locator (RLOC) [4].

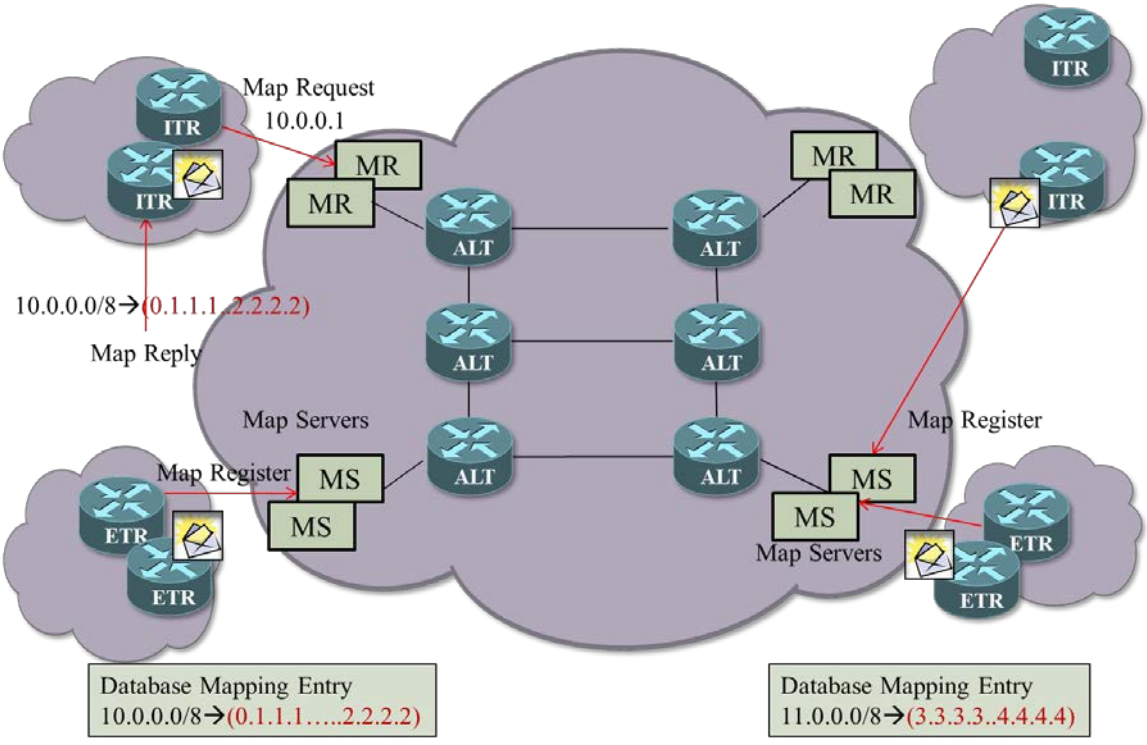

Figure 13: LISP+ALT Architecture

As shown in the above Fig. 13; LISP+ALT is made up of ALT Routers build as an overlay network over the public Internet which are interconnected through tunnels namely “Generic Routing Encapsulation” (GRE), where each of these ALT-routers use

“Border Gateway Protocol” (BGP) to propagate path information needed to route the packets known as ALT Datagram [4], which is basically a LISP control packet with Map-Request information.

Each of the above ALT-routers are deployed in a “Hierarchical-Mesh” Network where routers at each level in the topology is responsible for both “Aggregation and Advertising” of EID-Prefixes learned from the router below them to the routers above them respectively [4]. The “edge” ALT-Routers is usually statically connected to Map-Resolvers and Map-Servers or in a rare case is statically connected to “edge” xTRs.

Below steps shows a typical role of the LISP+ALT architecture aiding an ITR to obtain EID-RLOC mapping information from the respective ETR:

1. A host “S” trying to establish a connection to a host “D” (D.ieee.com) at the destination LISP-site, sends an IP packet to one of its assigned ITRs at the Source LISP-site, respective ITR (ITR-1) does a mapping looking and fails to find an EID-RLOC mapping entry to the destination host ‘D’ with IP address 10.0.0.1.
2. ITR-1 builds a LISP-Control packet with destination-EID prefix 10.0.0.1 and sends this Map-Request packet called “ALT Datagram” [4] to its associated Map-Resolver (MR).
3. The MR forwards this ALT Datagram to its statically connected ALT Router, this “first-hop” ALT Router looks up its “ALT BGP Route Information Base” which is comprised of EID-Prefixes and associate next hope ALT Routers [4]; and forwards the ALT Datagram to its next hope ALT Router, which in turn routes the packet via BGP to the “MS” which initially advertised this prefix.

4. Once the ALT Datagram reaches the associated “MS”, the “MS” forwards this packet to the appropriate ETR (ETR-2) which “owns” this prefix.
5. The ETR-2 treats the ALT Datagram as a Map-Request message and replies with a Map-Reply message that lists the RLOCs to the specific ITR (ITR-2).

2.5 Packet Flow Sequence b/w Two LISP Sites

In this section we will provide an example of “Unicast Packet Forwarding” [2] between two LISP enable sites with an assumption that ITR’s at each site already has the required mapping entry (EID-RLOC Mapping) for their respective EIDs.

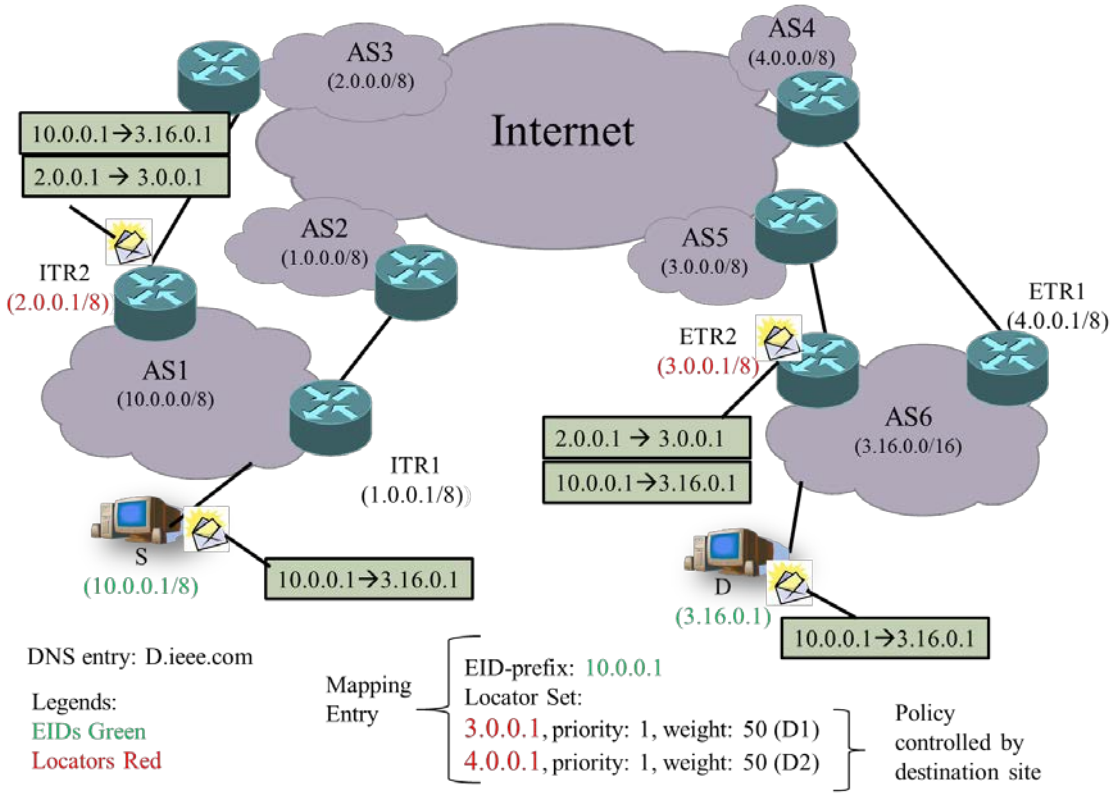

Figure 14: Unicast Packet Forwarding Between Two LISP Sites

In the figure above both sites EIDs have Provider Independent IP prefixes and the respective xTRs with routable IP addresses connect to the upstream provider networks. When the host 'S' (s.umkc.edu) with EID-prefix of *10.0.0.1* wants to communicate with the host at the destination site 'D' (D.IEEE.com) it follows the below steps

1. Host 'S' (s.umkc.edu) does a DNS lookup on D.ieee.com and obtains an IP address of 'D' as *3.16.0.1*.
2. Host 'S' prepares an IP packet with IP address *10.0.0.1* as the Source-EID and IP address *3.16.0.1* as the Destination-EID and forwards this packet to one its assigned ITRs (ITR-2).

3. When the packet reaches the ITR-2, it checks its map-cache entry for the EID *3.16.0.1* and obtains its RLOC mapping entry as shown in the above Fig. 14 with the label “Mapping Entry”.
4. Based on the above “Mapping Entry” ITR-2 encapsulate the IP packet with the an outer LISP-Header where the source-RLOC IP address is *2.0.0.1* and destination-RLOC IP address will be *3.0.0.1* and forwards the packet to the upstream provider, which routes the packet in the Internet based on the outer LISP header destination address.
5. As the packet reaches ETR-2, it strips the outer LISP-header and based on the destination-EID’s IP address from the inner IP header forwards the packet to the host D.ieee.com.

2.6 Advantages of LISP

This section identifies some of the advantages of the Locator/Identifier Separation Protocol with respect to its ability to significantly minimizing the scalability problem by reducing the routing table size, LISP solution for its incremental deployment and most importantly its comparison to BGP regarding exploiting the “path-diversity” the Internet provides for better Traffic Engineering practices without adding to the scalability issues.

2.6.1 LISP Impact on Routing Table Size

Increases in the routing table size (RIB) because of injection of unwanted prefixes into the DFZ will impact the performance of these routers in processing

incoming packets thereby introducing delays between endpoints. In this section we will explore how the Locator/Identifier split architecture that LISP provides wherein only RLOCs are routable and hence advertised in to the DFZ thereby considerably reducing the routing table size.

For Example, let us assume that there are 100,000 prefixes over 5 networks, under the legacy architecture each network will advertise 100,000/5 = 20,000 prefixes assuming equal load [16]. Now, assuming that LISP has been deployed and that for every 10 EID prefixes there is 1 RLOC associated, then, each network will have to advertise only 20,000/10 = 2,000 RLOCs instead of 20,000 prefixes. A simple equation for the “Number of Prefixes Advertised” [16] can be devised as below:

$$\text{Number of Prefixes Advertised} = \frac{(\#\text{ of Prefixes})/(\#\text{ of Networks})}{(\#\text{ of Prefixes per RLOC})}$$

Below picture and table demonstrates the impact of LISP on reducing the Routing Table Size.

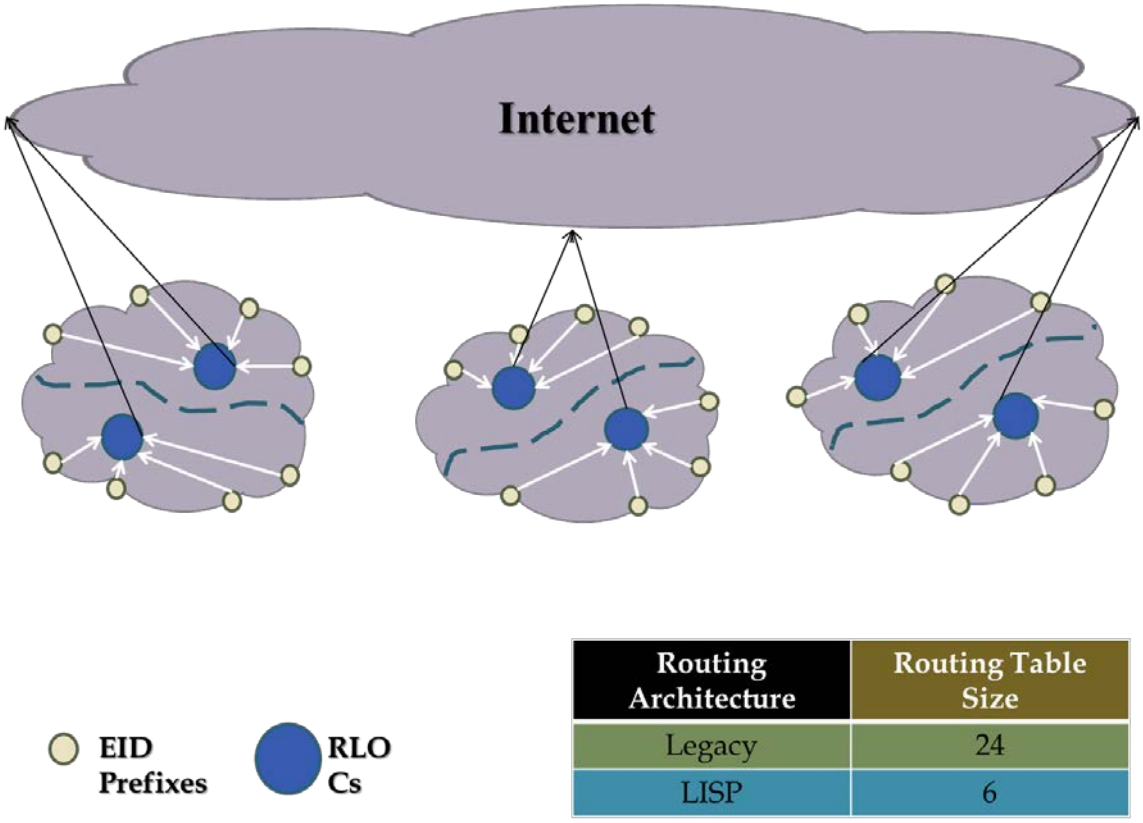

Figure 15: Routing Table Size (LISP v/s Legacy Architecture)

Table 6: Shrinking the Router Table Size

# of Prefixes	# of N/Ws	Legacy Table Size	LIPS Deployed Table Size(Prefixes/ RLOC)	
			10 (Prefixes)	20 (Prefixes)
30,000	2	15,000	1500	750
	5	6,000	600	300
	10	3,000	300	150
40,000	2	20,000	2000	1000
	5	8,000	800	400
	10	4,000	400	200

2.6.2 LISP Support for Incremental Deployment

To facilitate “incremental deployment of LISP” it is imperative that LISP addresses interoperation of LISP enabled (EIDs) and non-LISP sites (Internet Sites with traditional IPv4 and/or IPv6 addresses) as even though there is no syntactical difference between an EID and an IP address but the way EIDs are routed in the global routing system is completely different [3] from the current practice hence the need for interoperation. In this section we look into two such mechanisms that LISP provides to address interworking of LISP with IPv4 and/or IPv6.

2.6.3 Interworking of Non-LISP and LISP Sites Using Proxy Ingress Tunnel Router (PITR)

Proxy Ingress Tunnel Router (PITR) as the name suggests act as a Proxy ITR when a non-LISP site wants to send packets to LISP enabled site. PITR has 2 main functions

- Initiating EID Advertisement: Because EIDs are non-routable in the Internet PITRs advertise extensively aggregated EID-prefix space on behalf of LISP sites thus aiding the non-LISP site to reach those [3].
- Encapsulating Legacy Internet Traffic: Facilitates encapsulation of legacy IPv4/IpV6 packets originating from the non-LISP sites into LISP packets and directs the packets towards the respective RLOCs.

2.6.3.1 Packet Flow in Presence of PITRs

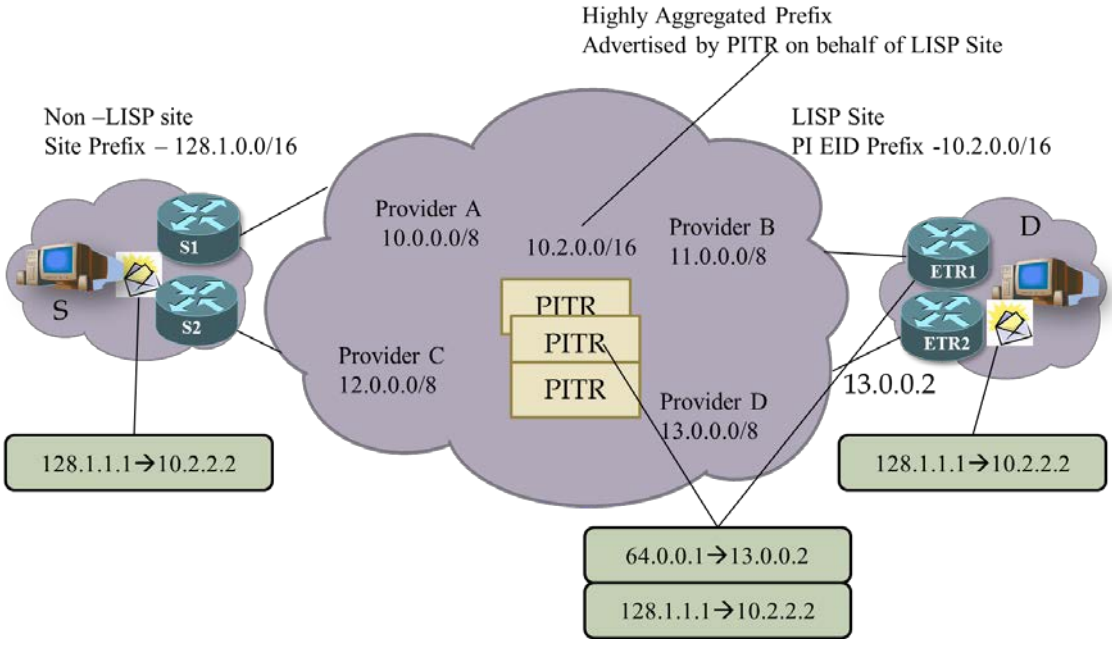

Figure 16: Interworking with PITRs

The above figure shows an example of a typical packet traversal between a Non-LISP site and LISP enabled site in presence of PITRs. The steps are as below:

1. The source node *(128.1.1.1)* at the Non-LISP site does DNS look-up and obtains the IP address *10.2.2.2*, note that the densely aggregated prefix *10.2.0.0/16* is advertised by PITRs on behalf of the LISP site.
2. The source node *(128.1.1.1)* routes the packet through it Customer Edge router through a default route which in turn routes the packet to the Provider Edge (PE) router, where PE has a route to reach the respective PITR.
3. Once the packet reaches the PITR, it obtains a EID-RLOC mapping either through Map-Request or from the local mapping cache and encapsulates the legacy IP packet with the LISP packet where, the inner header has the PITR IPv4 address *(10.2.2.2)* as the destination address and the outer LISP header has the appropriate RLOC addresses, where IP address *(13.0.0.2)* is the destination RLOC address.
4. With this encapsulation the PITR routes the packet to the next hop router, after which, the packet is routed to the destination RLOC *(13.0.0.2)*.
5. When the packet reaches its respective destination RLOC *(13.0.0.2),* it de-encapsulates the outer LISP header and routes the packet internally to the destination EID *(10.2.2.2)*.
6. Packets from destination EID *(10.2.2.2)* going back to source node *(128.1.1.1)* will flow through the LISP-Site ITR but at the ITR these packets are not encapsulated as the destination nodes IP address (*128.1.1.1)* is globally routable.

2.6.4 Interworking of LISP Sites and Non-LISP Sites Using Proxy Egress Tunnel Router (PETR)

Proxy Egress Tunnel Router allows communication between a LISP Site and a Non-LISP Site but before understanding the working on PETR, below we identify the importance or need for such a new network element to facilitate interworking of LISP.

2.6.4.1 Importance/Need for PETR

In today's world Security is the most important aspect of any infrastructure and hence all the Providers Edge (PE) routers are inundated with Access Control Configurations, among which, one is very prevalent called Unicast Reverse Path Forwarding (uRPF) rule [3]. uRPF rule basically states that if an incoming packet's source IP address is not recognizable (globally routable) you simply drop those packets. Since in our LISP topology EIDs are non-routable and hence are not advertised to the outside world will suffer from same fate as ITRs at the LISP site when sending packet to a non-LISP sites do not encapsulate the IP header with LISP header. PETR provides a solution to this problem by bypassing the uRPF check at PE routers by encapsulating all the LISP sites egress traffic with LISP header destined to Non-LISP site to them.

2.6.4.2 Packet Flow in Presence of PITRs

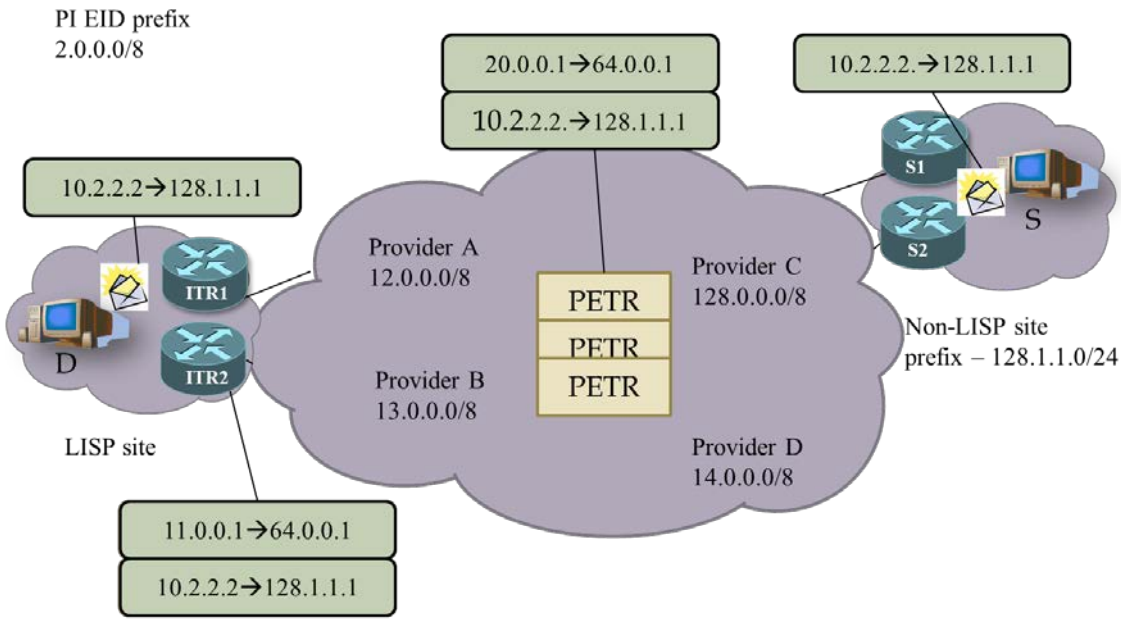

Figure 17: Interworking with PETRs

The above Fig. 17 shows an example of a typical packet traversal between a LISP site and a Non-LISP site in presence of PETRs. The steps are as below:

1. The source node *(EID-10.2.2.2)* at the LISP site does DNS look-up on the destination and obtains the IP address *128.1.1.1*.
2. The source EID (*10.2.2.2)* selects one of its corresponding ITRs and forwards the packet to ITR2, which happens to be the sites Customer Edge (CE) Router.
3. The ITR2 at the LISP site is been configured to encapsulate all the traffic going towards a non-LISP site with a LISP header and route it to a Proxy-ETR.
4. Once the packet reaches a corresponding PETR, it decapsulates the outer LISP header and routes the original packet to its next hop and from there the

packet is routed to the destination node *(128.1.1.1)* in the Non-LISP site natively.

CHAPTER 3

CURRENT TRAFFIC ENGINEERING PRACTICES

3.1 Need for Traffic Engineering

Today's Internet is basically a collection of distinct domains, where each domain corresponds to an Organization or an Tier-1 Internet Service Provider (ISP), we classify these domains in to either a Stub domain; which do not carry traffic that are not generated by and/or destined to their hosts and a Transmit domain; which acts as a bridge carrying traffic generated by and/or destined to external domains [7].

The need for Traffic Engineering comes into picture because of the need to run or satisfy mission critical applications with stringent SLA's over the "best-effort service" model [7] that our Internet provides. Network Engineers look to reduce the delay or congestion using Traffic Engineering techniques which can be classified as "Outbound Traffic Engineering" (OTE) and "Inbound Traffic Engineering" (ITE). OTE dictates controlling the traffic going out from a domain, where they can choose to tune intra-domain routing protocols like OSPF or EIGRP to better utilize the network if load balancing is an optimization criterion or you can use techniques like MPLS to reduce the latency by eliminating costly route look-up at individual routers. Apart from optimizing the flow of packets inside their own network, sometimes it becomes imperative to control the flow of packets coming into their network which we refer to ITE. To achieve ITE the only tool available today is by tuning the inter-domain protocol called Border Gateway Protocol (BGP) [7].

3.2 Inter-Domain Traffic Engineering Using BGP Attributes as a Metric

In this section we explore current inter-domain traffic engineering practices using BGP and their limitations, including wide spread practices like “Selective Advertisement” promoting local benefit at a global cost and thereby directly influencing the Internet’s scalability issues.

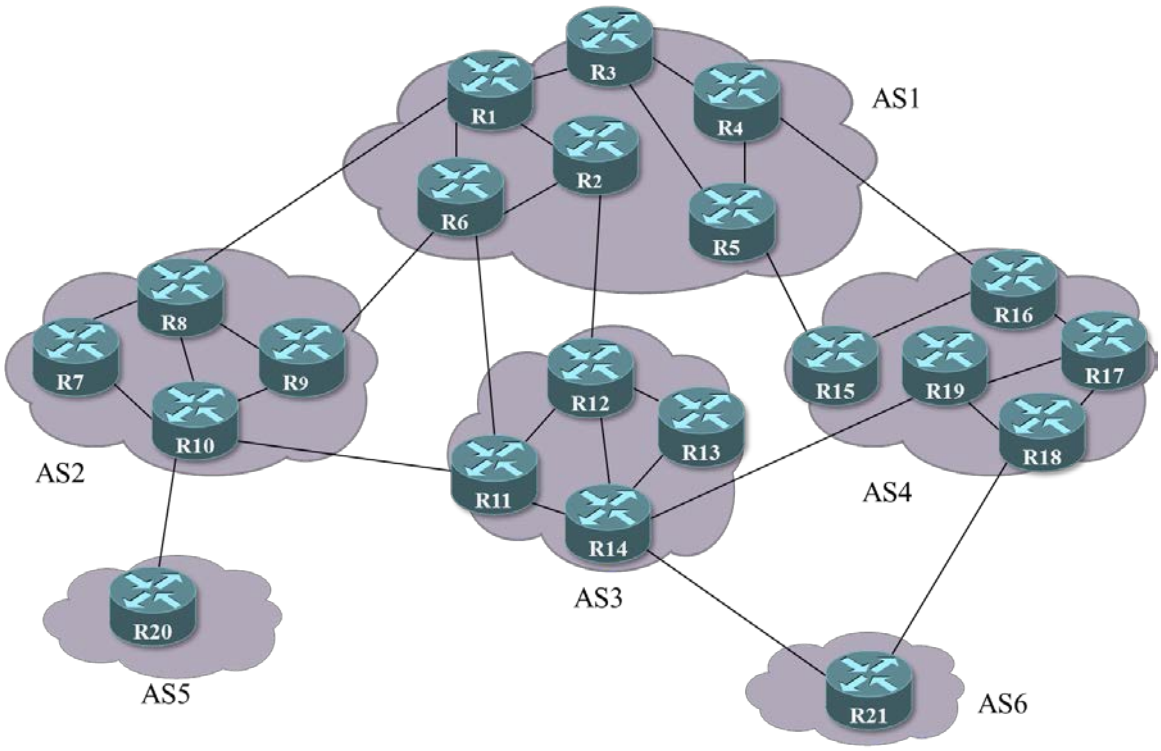

Figure 18: Transit and Stub ASes Forming a Simple Internet

Some of the BGP mandatory attributes like “AS-Path” and optional attributes like “Multi-Exit-Discriminator”, “Redistribution Communities” can be used as a metric to facilitate Inter-Domain Traffic Engineering as discussed below:

- AS-Path

AS-Path attribute is exchanged in the BGP Update message between two BGP speaking routers, where in, this attribute stores a sequence of Autonomous (AS) Numbers identifying the ASes a route has visited so far [5].

AS-Path as a traffic engineering metric comes into picture when a source AS evaluates the distance to one if its destination ASes with respect to number of hops based on the length of the AS-Path attribute it receives in the Update messages from its neighbors. Given this situation a transit AS can control the flow of packets coming in to its network by manipulating the length of the AS-Path by prepending its own AS number more than once in the AS-Path attribute and thus indicating a ranking among the various route advertisement that it sends to its peers [7]. An example is as shown below:

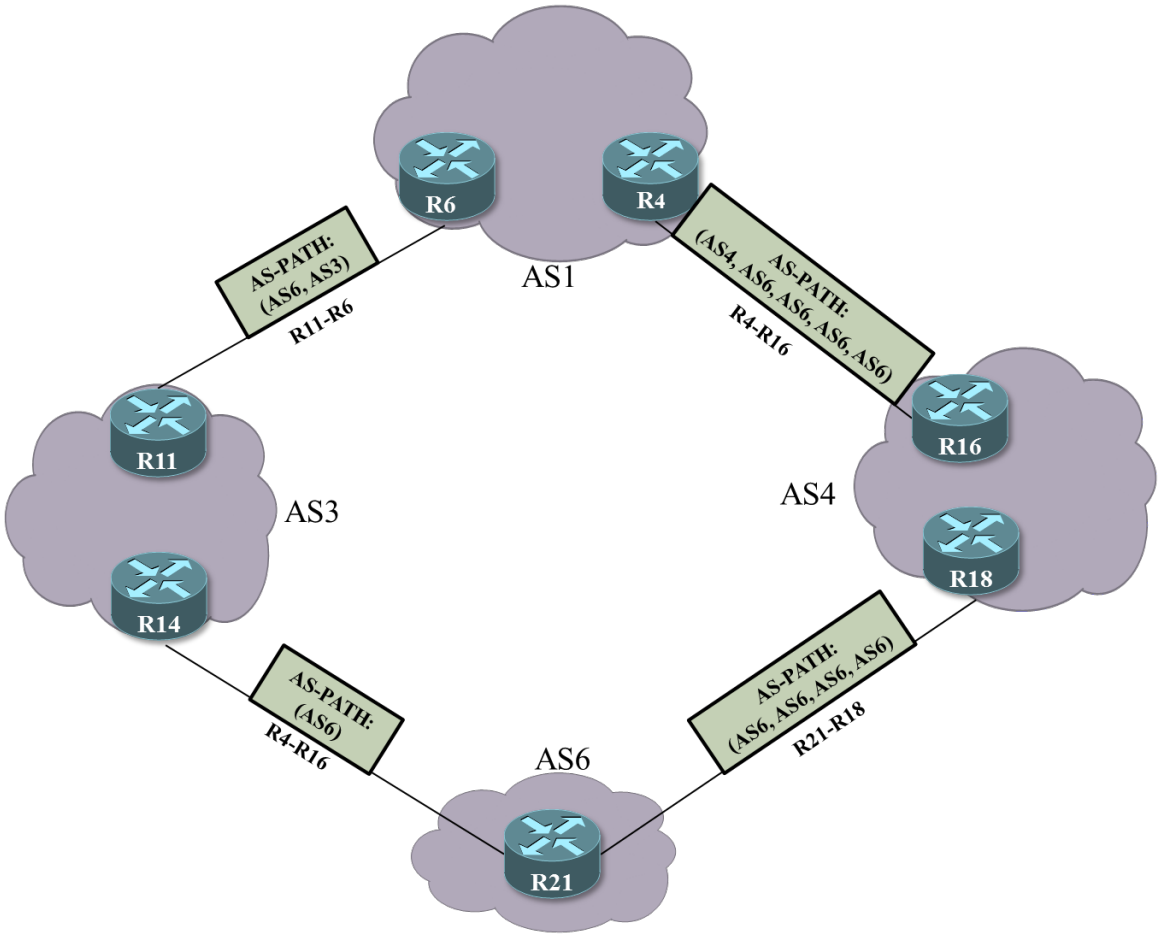

Figure 19: AS-Path advertisement from AS6

From Fig. 19; AS6 has two inter-domain links one connected to AS3 and other to AS4, assuming that AS6 wishes to allocate link R21-R14 as the primary inter-domain link and the link R21-R18 as the backup primary inter-domain link, it can achieve this by advertising the routes on the primary link R21-R14 with AS-Path attribute of (AS6) and artificially increasing the AS-Path attribute length as when advertising the route on the backup primary link R21-R18. Thus, the route advertised on the primary inter-domain link would be considered as the best route by the routers which do not rely on manually configured settings for the *weights* and *local-pref* attributes [7] and thereby forcing these routers to send and receive traffic on the primary inter-domain link R21-R14.

- Multi-Exit-Discriminator (MED)

MED is an optional attribute which can be used as a metric only if an AS has multiple external links to its neighbors as shown below in Fig. 20.

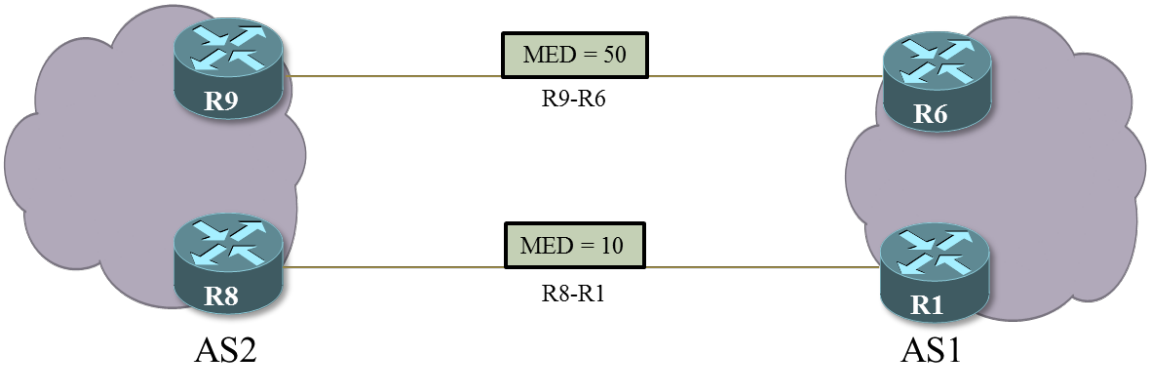

Figure 20: Multi-Exit-Discriminator as an Inter-Domain TE Metric

From Fig. 20: AS2 has two external links connecting to AS1, if AS2 wants to use link R8-R1 to control the traffic coming from AS1, it can achieve this by decreasing the MED value for the link R8-R1 compared to the link R9-R6 and thus forcing the AS1 to use only the link R8-R1 for communication.

- Redistribution Communities

A redistribution community is an optional attribute which can be attached to routes for traffic engineering purposes. The redistribution communities attached to the route defines both the traffic engineering action to be performed and the BGP peers that are affected by this action [7], one such action is for an AS to request its upstream peers to perform AS-Path prepending when redistributing the routes to the specified peers, example is as shown below:

Suppose AS6 in Fig. 21 receives lot of traffic from both AS1 and AS2, so to manage the load it wants to utilize both of its external links connected to peers AS3 and AS4 to help share the traffic entering into its domain. AS6 cannot achieve such a traffic distribution by itself using AS-Path prepending technique [7]. However, using redistribution communities it can request the upstream AS3 to perform AS-Path prepending (Action) when redistributing the route to AS2 and AS1 (BGP Peers).

Thus, AS3 when redistributing the route to AS1, it artificially increases the AS-Path attribute by prepending (AS3 AS3 AS6) and advertises normal AS-Path (AS3 AS6) to AS2. Thus, AS2 uses the shorter route of AS3→AS6, thus reaching AS6 through link R21-R14 and similarly AS1 uses the shorter route AS4→AS6 instead of AS3→AS3→AS6, thus reaching AS6 through the link R18-R21.

Table 7: Redistributing Community PREPEND Values

Target AS	Upstream AS	Redistribution-Community Values	Traffic Engineering Purpose
AS6	AS3	Community: PREPEND Action: AS-Path Prepending BGP Peers: AS1 And AS2	Load Balancing through links R21-R14 and R-21-R18

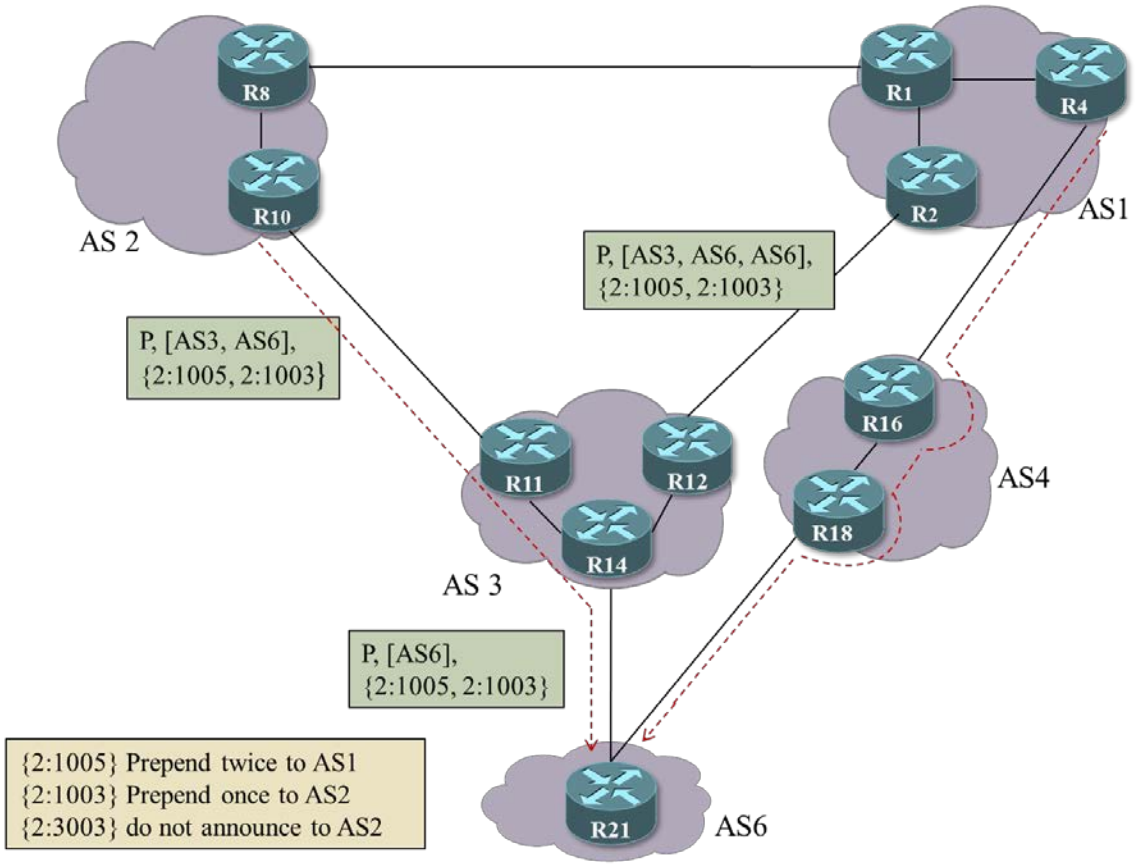

Figure 21: Community-Based Traffic Engineering

3.3 Limitations of BGP Attribute Based Inter-Domain Traffic Engineering

An AS can use BGP attributes as a metric to control the flow of packets between its peers for different optimization purpose but each of the technique discussed above have serious limitations. Firstly, to use "Multi-Exit-Discriminator" as a traffic engineering metric an AS should have more than one external links connecting to its individual peers which might not be true in all the cases. Secondly, neither AS-Path prepending nor redistribution communities are useful if the sources from which an AS wants to control the traffic coming into it is attached to the same provider [6] as shown below in Fig. 22.

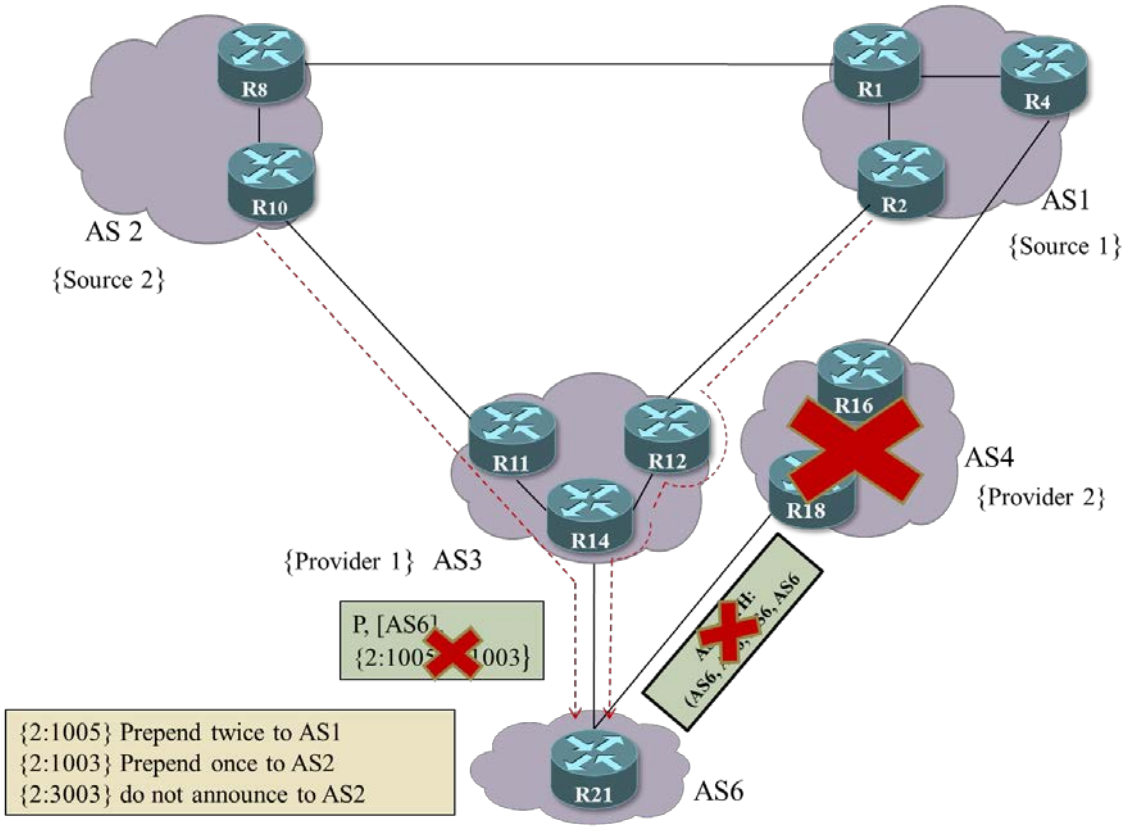

Figure 22: Limitations of AS-Path Prepending and Redistribution Communities

3.4 Polluting the Internet: Toxic Inter-Domain Traffic Engineering Practices

Because of the limitations posed by the BGP attributes when used as a traffic engineering metric and the need for finer control on redistribution of routes, ISP's are tend to opt for traffic engineering practices like "Selective Advertisements" and "Advertisement of more Specific Prefixes" to gain local benefit at a global cost, explained as below

- Selective Advertisement

An AS which wants to impose stringent policies to control the traffic entering into its domain may opt for selective advertisement; which is nothing but advertising different routes on different links. For example, in Fig. 18, if transit AS; AS2 has load balancing as an optimization criterion over the links R8-R1 and R9-R6, it can achieve this by announcing only its internal routes via the link R8-R1 and the routes learned from the stub AS; AS5 via link R9-R6 [6]. Since, AS1 learns about AS5 only through the router R6; it will use only the link R6-R9 to send any traffic destined for the stub AS; AS5.

- Advertising More Specific Prefixes

  Today's IP routers live and breathe on the fact that it always selects from its forwarding table the most specific route (matching route with the longest prefix) for each packet. ISP's tend to use this fact as a vantage point and try to control the packets entering into their domain by advertising the more specific prefixes as shown below.

From Fig. 18; suppose AS3 is a major Content Delivery Network (CDN) and as a result hosts many servers in its domain. Also, assuming that the aggregate IP prefix of AS3 is 112.0.0.0/8 and the subnet which hosts all of the major CDN servers is 112.10.11.0/24.If AS3 prefers to receive the request for content on one of its link R11-R6; then it would advertise the more specific prefix on the link R11-R6 and less specific prefix on link R12-R2, thereby forcing all the incoming requests to come through link R11-R6 and using link R12-R2 as backup/restoration purpose.

3.4.1 A Major Drawback

As you can observe from the above discussion that the techniques used by AS in order to control incoming traffic will result in advertising more unwarranted prefixes into the DFZ, all these prefixes will be propagated throughout the global internet thus increasing the size of BGP routing table of almost all ASes and thus directly influencing the scalability issues of today's Internet.

CHAPTER 4

RLOC-DRIVEN TRAFFIC ENGINEERING IN A LISP NETWORK

LISP separation of single numbering space, namely the *IP addressing* where an IP address is used for both host transport session identification and network routing provides opportunities to explore path diversity inherently present in today's Internet by associating a single EID with multiple RLOCs, this association offers a new dimension to inter-domain traffic engineering and makes it possible to choose a best route to a locator based on some optimization criterion like delay/latency or facilitate traffic proportioning (load-balancing) in presence of multiple locator sets (RLOCs).

The models discussed here are presented as part of the joint work [14] and are reproduced in this thesis for completeness and in order to present and discuss the results in the subsequent chapter.

4.1 Scope of Our Work

In this work we mainly address the advantages for flexible inter-domain traffic engineering that the LISP offers in presence of multiple-RLOCs with traffic proportioning or load-balancing as the optimization or performance criterion. Here, we define a notion of a *"group"* in a LISP network; when we say LISP network it is that portion of the figure marked with dotted oval in Fig 23; as below. Before defining the notion of a group we divide all routers in a LISP network in to "Regular Routers" and

"RLOC Routers", by RLOC routers, we mean the routers belonging to one or more groups for traffic proportioning.

A group is a collection of two or more routers that are in geographic proximity to each other and may be associated with an ETR for traffic proportioning. One such group can be RLOCs (RLOC2A, RLOC2B) in Fig. 23; Two cases are considered, where, in the first case one of the RLOCs in the group will serve as a "Primary RLOC"; primary destination for the LISP packet at the destination site, if traffic proportioning between multiple RLOCs is not initiated, the remaining routers in the same group would act as a "Secondary RLOCs/Routers". In the second case, all of the RLOCs in a group may serve as either "Primary" or "Secondary" RLOCs; i.e., traffic destined for any RLOCs in the group may be split among its peers within the same group, with no distinction of a single RLOC as the "Primary RLOC".

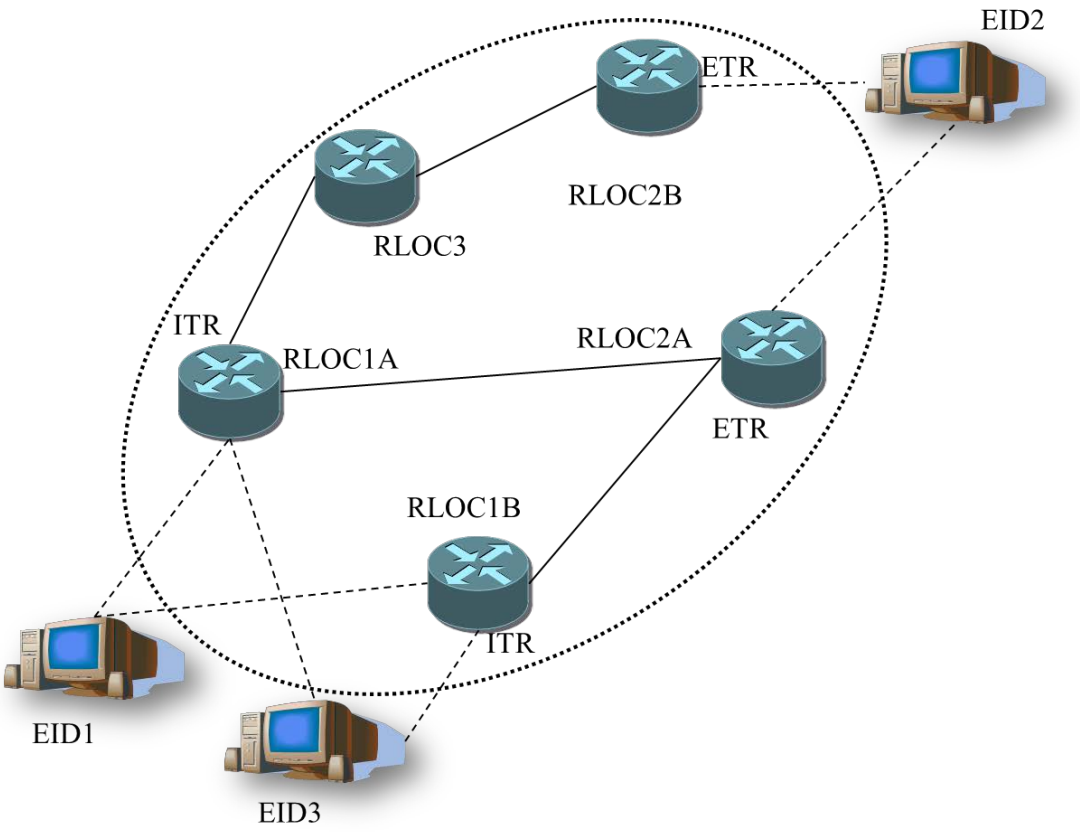

Figure 23: An Example of LISP-Enabled Network

After the optimization models, we present the computational results showing the advantages or effectiveness of LISP enabled traffic engineering compared to the base case which identifies the limitations of today's routing architecture by restricting traffic proportioning to multiple RLOCs.

The rest of this chapter is organized as follows: in Section II, LISP-Enabled Traffic Engineering Formulations are presented. In section III, we present our computational results considering topologies namely, Internet2, AboveNet and Exodus; and identify the advantages that LISP-enabled traffic engineering offers compared to the Base-TE traffic engineering. And, finally, in Section IV we present our conclusion and scope for future work.

4.2 LISP-Enabled Traffic Engineering Formulation

It is assumed that the required coordinated routing mechanism between multiple autonomous systems involved between source and destination sites is already in place to reap the benefits of traffic engineering in a LISP-context and the details on how these autonomous systems play a part is deemed out of scope, thus focusing on understanding the benefits of flexible traffic engineering architecture that LISP provides with network resource utilization (Load-Balancing) as an optimization criterion.

4.2.1 Three Cases Considered For Traffic Engineering

Case-I: Base-TE

In this case, there is no notion of a group i.e. the traffic destined for any one of the router in a group is not load balanced between its peers in that same group, this case is a typical example of current routing architecture where the BGP selects only the best path to its destination using longest matching prefix rule. For example, in Fig. 23; given a group with routers (RLOC2A, RLOC2B) according to Base-TE (Case-I), any traffic destined for RLOC2A will not be proportioned to RLOC2B and vice-versa.

Case-II: LISP-TE II

In this case, one of the RLOCs in any given group is identified to be a “Primary” router (for destination) and its peer RLOCs are identified as “Secondary” routers in the same group. With this distinction any traffic destined for a primary RLOC may be load-balanced/proportioned among its Secondary RLOCs but not the other way around. For example, in Fig. 23; given a group with routers (RLOC2A, RLOC2B) with RLOC2A designated as a primary RLOC according to LISP-TE II (Case-II), any traffic destined to RLOC2A may be load-balanced/proportioned to RLOC2B but traffic destined to RLOC2B will not be load-balanced/proportioned back to RLOC2A.

Case-III: LISP-TE III

In Case III, all RLOCs in a group are identified as a “Primary” router (for destination) and the traffic destined for any router in the group may be load-balanced/proportioned to its peers in the same group. For example, in Fig. 23; given a group with routers (RLOC2A, RLOC2B) with according to LISP-TE III (Case-III), any

traffic destined to RLOC2A may be load-balanced/proportioned to RLOC2B and vice-versa.

4.2.2 Notations and Variables

Notations are detailed in Table 8. Group identification is done through one of its member routers (RLOCs) i.e. if router d belongs to a group then G^d reflects its group. For example if RLOCs 1 and 2 belongs to a group say Group-A and RLOCs 9 and 10 belongs to a group say Group-B, then G^1 or G^2 will represent Group-A and G^9 or G^{10} will represent Group-B. The notation W is the super set of the individual set of routers in a group (G^d).i.e. W = {{1, 2}, {9, 10}}. Also we define G* to be the set of individual elements of RLOCs who are part of any group (G^d) i.e. G* = {1, 2, 9, 10}. Thus in a set representation $\bigcup G^d = W$, where $d \in G^*$.

The set of primary RLOCs belonging to different groups are identified using $\widehat{R}$. Finally, we assume that groups are disjoint and not intersecting i.e. a RLOC in one group is not present in any other group.

Table 8: Lists of Notations and Variables

Given
S = Set of source routers
D = Set of destination routers
G^*= Set of all routers belonging to all the groups of routers for traffic proportioning
G^d = Set of all routers in a group where router d belongs in, usually destination router d

W = Union set of the individual set of groups G^D

L = set of links in the network

$\hat{R}$= Set of primary routers from group routers

h_{sd}= traffic demand between s and d

C_{lm} = capacity of link l-m

P_{sd} = Set of paths from source router s to destination router d

$\delta^p_{sd,lm}$= link-path indicator, set to 1 if path p for demand between source s and destination d uses link l-m; 0, otherwise

Variables

x^p_{sd}= non-negative flow variable s to d on path p

z^p_{st}= non-negative flow variable z from s to d path p when d is a secondary router

α_{sd} = fraction of demand between source and primary router where d is a primary router ($0 \leq \alpha_{sd} \leq 1$)

α^t_{sd} = fraction of demand between source s and primary router d sent to secondary router t in the same group ($0 \leq \alpha^t_{sd} \leq 1$)

y_{lm}= link flow variable for link l-m

r = maximum link utilization variable

4.2.3 LISP-enabled Traffic Engineering Formulation for LISP-TE II

Demand Constraints	
$\sum_{p \varepsilon P_{sd}} x_{sd}^{p} = h_{sd} \quad s \varepsilon S, d \varepsilon D \backslash \hat{R}, s \neq d$	(1)
$\sum_{p \varepsilon P_{sd}} x_{sd}^{p} = h_{sd}\, \alpha_{sd} \quad s \varepsilon S \backslash G^{d}, d \varepsilon \hat{R}$	(2)
$\sum_{p \varepsilon P_{st}} z_{st}^{p} = h_{sd}\, \alpha_{sd}^{t} \quad s \varepsilon S \backslash G^{d}, t \varepsilon G^{d} \backslash \{d\}, d \varepsilon \hat{R}$	(3)
Routing Restriction	
$\alpha_{sd} + \sum_{t \varepsilon R_{sd}} \alpha_{sd}^{t} = 1 \quad s \varepsilon S \backslash G^{d}, d \varepsilon \hat{R}, t \varepsilon G^{d} \backslash \{d\}$	(4)
Capacity Constraint	
$\sum_{s \varepsilon S, d \varepsilon D \backslash \hat{R}, s \neq d} \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} x_{sd}^{p} + \sum_{s \varepsilon S \backslash G^{d}, d \varepsilon \hat{R}} \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} x_{sd}^{p} + \sum_{s \varepsilon S \backslash G^{d}, d \varepsilon \hat{R}} \sum_{t \varepsilon G^{d} \backslash \{d\}} \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} z_{st}^{p} = y_{lm} \; (l,m) \varepsilon L$	(5)
Objective Function	
Minimize r $y_{lm} \leq C_{lm} r \; (l,m) \varepsilon L$	(6)

4.2.3.1 Demand Constraints in LISP-TE II

A demand constraint is a constraint where demand volume/traffic flow for each demand from source to destination needs to be realized through flows on candidate paths [11]. Demand constraints in LISP-TE II can be divided into three situations.

$$\sum_{p \varepsilon P_{sd}} x_{sd}^{p} = h_{sd} \quad s \varepsilon S,\ d \varepsilon D \backslash \hat{R},\ s \neq d \qquad (1)$$

Equation (1) represents a traffic flow or demand volume between a source router s and destination router d over a path p, where destination router d does not belong to a set of primary routers ($\hat{R}$), so they can belong to the secondary RLOCs in a group or to any routers at the destination site which is not a part of any group. This is because the demand destined for the secondary RLOCs in a group is not proportioned with its peers in the same group in LISP-TE II case.

$$\sum_{p \varepsilon P_{sd}} x_{sd}^{p} = h_{sd}\, \alpha_{sd} \quad s \varepsilon S \backslash G^{d}, d \varepsilon \hat{R} \qquad (2)$$

Equation (2) represents a traffic flow or demand volume between a source router s; where the source router belongs to the routers outside the group of primary routers and a destination router d; where the destination router belongs to a set of primary RLOCs, over a path p. The new variable (α_{sd}) represents the proportion of the traffic flow towards the destination primary RLOC in a given group, where, the remaining proportion (fractioned by) of the total demand volume or traffic flow is proportioned to its peers (secondary RLOCs) within the same group.

$$\sum_{p \varepsilon P_{st}} z_{st}^{p} = h_{sd}\, \alpha_{sd}^{t} \quad s \varepsilon S \backslash G^{d}, t \varepsilon G^{d} \backslash \{d\}, d \varepsilon \hat{R} \qquad (3)$$

In Equation (3), a new flow variable z_{sd}^{p} is introduced, which is the proportioned flow of primary RLOC to its secondary RLOCs with in the same group i.e. the demand it carries is for the primary RLOC but the path it takes is for the secondary RLOCs within the same group.

$$\alpha_{sd} + \sum_{p \varepsilon P_{sd}} \alpha_{sd}^{t} = 1 \quad s \varepsilon S \backslash G^{d}, d \varepsilon \hat{R}, \; t \varepsilon G^{d} \backslash \{d\} \qquad (4)$$

Equation (4) represents the "Routing Restriction" i.e. the proportion of traffic towards primary RLOC (α_{sd}) and (α_{sd}^{t}) the remaining portion of the traffic proportioned to its associated secondary RLOCs within the same group must add up to 1.

4.2.3.2 Capacities Constraints in LISP-TE II

Capacity constraints are a set of constraints which assures that for each link (*l*, *m*) (connecting nodes *l* and *m*) the flow on that link y_{lm} (which accounts for all the traffic flow variables using that link) should be less than or equal to the given capacity of that link $C_{lm.}$ Thus, the flow on a link is given by:

$$\sum_{s \varepsilon S, d \varepsilon D \backslash \hat{R}, \; s \neq d} \; \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} x_{sd}^{p} + \sum_{s \varepsilon S \backslash G^{d}, d \varepsilon \hat{R}} \; \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} x_{sd}^{p} + \sum_{s \varepsilon S \backslash G^{d}, d \varepsilon \hat{R}} \; \sum_{t \varepsilon G^{d} \backslash \{d\}} \; \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} z_{st}^{p} = y_{lm} \quad (l,m) \varepsilon L \qquad \textbf{(5)}$$

Here, $\delta_{sd,lm}^{p}$ is a link-path identifier where,

$\delta_{sd,lm}^{p}$ = 1, if the flow variable between s and on path p exits on link (*l*, *m*)

0, Otherwise

4.2.3.3 The Objective Function for LISP-TE II

The objective of our formulation is to optimize network utilization by minimizing the maximum link utilization. Where, *r* represents the maximum link utilization variable which relates to the link capacity with link flow on (*l*, *m*) as below:

Main Objective is:

$$\text{Minimize } r \qquad (6)$$

$$\text{Where, } y_{lm} \leq C_{lm} r \ (l,m) \varepsilon L$$

4.2.3.4 Base-TE: A Special Case in LISP-TE II

With respect to above model, the Base-TE (Case I) where traffic is not proportioned to RLOCs in a group becomes a special case, when we change the "Routing Restriction" in equation (4) is changed by setting $\alpha_{sd} = 1$, which implies $\alpha_{sd}^{t} = 0$, $s \varepsilon S \backslash G^{d}, d \varepsilon \hat{R}, t \varepsilon G^{d} \backslash \{d\}$.

4.2.4 LISP-Enabled Traffic Engineering Formulation for LISP-TE III

Similar to LISP-TE II, the traffic engineering formulation for LISP-TE III is discussed. The main difference in LISP-TE III formulation is that the traffic is proportioned to all the RLOCs in a group as opposed to proportioning it from the primary router to the secondary routers in a group. That is, there is no notion of primary RLOCs or secondary RLOCs in LISP-TE III; traffic destined for any RLOCs belonging to any group is proportioned among all its peers within the same group.

Demand Constraints	
$\sum_{p \varepsilon P_{sd}} x_{sd}^{p} = h_{sd} \quad s \varepsilon S, d \varepsilon D \backslash G^{*}, s \neq d$	(7)
$\sum_{p \varepsilon P_{sd}} x_{sd}^{p} = h_{sd}\, \alpha_{sd} \quad s \varepsilon S \backslash G^{d}, d \varepsilon G^{*}$	(8)
$\sum_{p \varepsilon P_{sd}} z_{sd}^{p} = h_{sd}\, \alpha_{sd}^{t} \quad s \varepsilon S \backslash G^{d}, t \varepsilon G^{d}, d \varepsilon G^{*}, d \neq t$	(9)
Routing Restriction	
$\alpha_{sd} + \sum_{t \varepsilon R_{sd}} \alpha_{sd}^{t} = 1 \quad s \varepsilon S \backslash G^{d}, t \varepsilon G^{d}, d \varepsilon G^{*}, d \neq t$	(10)
Capacity Constraint	
$\sum_{s \varepsilon S, d \varepsilon D \backslash G^{*}, s \neq d} \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} x_{sd}^{p} + \sum_{s \varepsilon S \backslash G^{d}, d \varepsilon G^{*}} \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} x_{sd}^{p} + \sum_{s \varepsilon S \backslash G^{d}, d \varepsilon G^{*}} \sum_{t \varepsilon G^{d} \backslash \hat{R}, d \neq t} \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} z_{st}^{p} = y_{lm} \; (l,m) \varepsilon L$	(11)
Objective Function	
Minimize r $y_{lm} \leq C_{lm} r \; (l,m) \varepsilon L$	(12)

4.2.4.1 Demand Constraints in LISP-TE III

Demand constraint in LISP-TE III can be divided into 3 cases as shown below:

$$\sum_{p \varepsilon P_{sd}} x_{sd}^{p} = h_{sd} \quad s \varepsilon S,\ d \varepsilon D \backslash G^{*},\ s \neq d \qquad (7)$$

Equation (7) represents a traffic/demand flow conservation equation for the general case, where the destination router d does not belong to any group.

$$\sum_{p \varepsilon P_{sd}} x_{sd}^{p} = h_{sd}\, \alpha_{sd} \quad s \varepsilon S \backslash G^{d}, d \varepsilon G^{*} \qquad (8)$$

Equation (8) represents the traffic flow to a destination RLOC which belongs to any RLOC in a group which may be spilt among its peers in the same group and proportioned to α_{sd}.

Equation (9) represents the remainder of the traffic fractioned by α_{sd}^{t} being proportioned to its peers in the same group.

$$\sum_{p \varepsilon P_{st}} z_{st}^{p} = h_{sd}\, \alpha_{sd}^{t} \quad s \varepsilon S \backslash G^{d}, t \varepsilon G^{d}, d \varepsilon G^{*}, d \neq t \qquad (9)$$

$$\alpha_{sd} + \sum_{t \varepsilon R_{sd}} \alpha_{sd}^{t} = 1 \quad s \varepsilon S \backslash G^{d}, t \varepsilon G^{d}, d \varepsilon G^{*}, d \neq t \qquad (10)$$

Equation (10) represents the "Routing Restriction" i.e. the proportion of traffic towards any destination RLOC (α_{sd}) and (α_{sd}^{t}) the remaining portion of the traffic proportioned to its peers within the same group must add up to 1.

4.2.4.2 Capacities Constraints in LISP-TE III

The Capacity constraint for LISP-TE III is similar to LISP-TE II, where link flow on link *(l, m)* accounts for the following equation

$$\sum_{s \varepsilon S, d \varepsilon D \backslash G^{*}, s \neq d} \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} x_{sd}^{p} + \sum_{s \varepsilon S \backslash G^{d}, d \varepsilon G^{*}} \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} x_{sd}^{p} + \sum_{s \varepsilon S \backslash G^{d}, d \varepsilon G^{*}} \sum_{t \varepsilon G^{d} \backslash \hat{R}, d \neq t} \sum_{p \varepsilon P_{sd}} \delta_{sd,lm}^{p} z_{st}^{p} = y_{lm} \quad (l, m) \varepsilon L \qquad (11)$$

4.2.4.3 The Objective Function for LISP-TE II

As in LISP-TE II, the objective for LISP-TE III is to minimize the maximum link utilization (*r*) and is given by as below

Main Objective is:

Minimize r (12)

Where, $y_{lm} \leq C_{lm} r \ (l,m) \varepsilon \ L$

4.3 Results and Topologies Considered

In order to evaluate the advantages that a LISP-enabled traffic engineering approach provides with respect to minimizing the maximum link utilization (*r*) of a network when compared to the base case (Base-TE) traffic engineering, we have conducted a number of studies with different topologies namely, Internet2, AboveNet and Exodus. All of our computational works are conducted using IBM ILOG CPLEX as the optimization tool.

4.3.1 Demand Generation

In our study we have considered both uniform and non-uniform demand and the demand generation model for the non-uniform case is as below [12]

$$\gamma O_x D_y C_{x,y} e^{-\beta(x,y)/2\Delta} \quad (13)$$

Where,

γ: is the scaling factor

O_x is a random number ε [0.1] with respect to router *x*.

D_y is a random number ε [0,1] with respect to router *y*.

$C_{x,y}$ is a random number ε [0,1] with respect to router *x* and *y*.

$\beta(x, y)$ is the Euclidean distance between router *x* and *y*.

$\Delta = \max\{x, y\}$; $x \varepsilon N$, $y \varepsilon N$, $x \neq y$, N is the set of all routers.

Note that we see the demand being generated is random and depends on the Euclidean distance between the routers and the demand set with respect to non-uniform case is the mean runs of equation (13).

4.3.2 Small Illustrative Topology

Here, in this section we begin with a small illustrative topology to discuss the salient features of LISP-TE model.

As shown in Fig. 24; consider a 5-node topology A5 with 5 routers. Let routers 2 and 5 be in a group, where, router 2 is a primary router and router 5 being a secondary i.e. G^2 = {2, 5} = G^5, W = {{2, 5}} and G^* = {2, 5}. Then, according to LISP-TE II, equation (4) reduces to the following three equations:

$$\alpha_{12} + \alpha_{12}^5 = 1 \qquad (14)$$

$$\alpha_{32} + \alpha_{32}^5 = 1 \qquad (15)$$

$$\alpha_{42} + \alpha_{42}^5 = 1 \qquad (16)$$

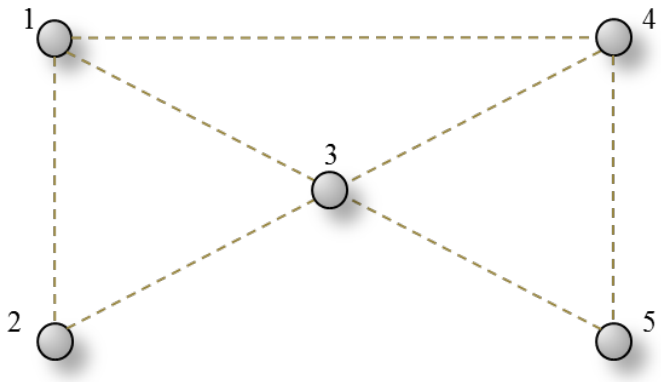

Figure 24: A Small 5-Node Topology (T5)

With respect to the above equations, we can strongly argue that the LISP-enabled traffic engineering (LISP-TE II) is beneficial only if we solve the LISP-TE II and find the solution to be $\alpha_{12} < 1$ or $\alpha_{32} < 1$ or $\alpha_{42} < 1$ but on the other hand if we find the solution to be $\alpha_{12} = \alpha_{32} = \alpha_{42} = 1$, then this would mean LISP-TE II has no gain or advantages over base case (Base-TE) traffic engineering. Note, that for this 5-node small topology we limit ourselves to LISP-TE II.

Below we present the results for two instances of traffic engineering, one with uniform demand and the other with non-uniform demands.

4.3.2.1 Uniform Traffic Demand with Uniform Link Capacities

As the name above this case represents a scenario where the traffic among all the routers is uniform, where we have set the demand to 175 and the capacities over all the links in the topology T5 is same; which is set to 400.

Table 9; below shows the results of the computation and we can observe that the maximum link utilization (*r*) in both the case is 72% and for LISP-TE II α_{12} = α_{32} = α_{42} = 1. This means that we do not see any benefits of LISP-TE II over the Base-TE. An important observation here is that the network symmetric around the group, and the traffic is also symmetric doing to uniform load, thus in this network, there is no benefit of multiple RLOCs from a traffic engineering perspective regardless of the load(in case of one fixed load).

Table 9: Uniform Demand, Uniform Capacity

(a) Base-TE

Capacity	r
400	0.72

(b) LISP-TE II

Capacity	r	α_{12}	α_{32}	α_{42}
400	0.72	1	1	1

4.3.2.2 Uniform Traffic Demand with Reduced Link Capacity

In this case we changed the capacity of a particular link (link 1-2) in the topology T5, to observe whether this will have any impact on traffic engineering. The Table 10 below shows the results for both Base-TE and LISP-TE II with different capacities for link 1-2. With respect to Base-TE, as the capacity decreases the optimal value of *r* increases as expected and becomes infeasible (r > 1) when the capacity is reduced to 100. Whereas, in LISP-TE II case, even though the optimal value of *r* increases with decrease in link capacity it happens at a slower pace. More importantly, we see that the value of α_{42} < 1, at reduced link capacity for link 1-2, thus as more and more traffic destined towards primary router 2 from 4 is diverted or proportioned to secondary router 5 and hence minimizing the maximum link utilization.

Table 10: Uniform Demand, Capacity adjusted on link 1-2

(a) Base Case

Capacity	r
300	0.75
200	0.87
100	1.05

(b) LISP-TE II

Capacity	r	α_{12}	α_{12}^5	α_{32}	α_{32}^5	α_{42}	α_{42}^5
300	0.72	1	0	1	0	0.91	0.08
200	0.75	1	0	1	0	0.57	0.42
100	0.80	1	0	1	0	0.30	0.69

4.3.2.3 Non-Uniform Traffic Demand with Uniform Link Capacity

Here, we consider non-uniform demands, where the traffic was generated using the demand generation formula in [12]. Table below shows the obtained demands, where the average demand is approximately 140. Again, the capacity was kept uniform throughout the network at 400.

Table 11: Non-Uniform Demand Generated

h_{12}=	h_3 =	h_{14}=	h_{15}=	h_{23}=	h_{24}=	h_{34}=	h_{35} =	h_{45} =
120	100	120	159	178	211	178	99	97

Below Table 12; shows the results of the computation, where the value of *r* remains the same for both Base-TE case and LISP-TE II case but we observe that traffic being proportioned/load-balanced between routers 2 and 5 for the demand flow between router 1 and 4 (α_{42} = 0.25 and α_{42}^{5} = 0.74), which we did not see with respect to uniform demand and uniform capacity. Thus, from this we can draw an important conclusion that with non-uniform demands and uniform link capacities, traffic proportion is possible as non-symmetry in traffic can influence the need for proportioning traffic to destinations.

Table 12: Non-Uniform Demand, Uniform Capacity

(a) Base Case

Capacity	r
400	0.64

(b) LISP-TE II

Capacity	r	α_{12}	α_{32}	α_{42}	α_{42}^{5}
400	0.64	1	1	0.25	0.74

4.3.2.4Non-Uniform Traffic Demand with Reduced Link Capacity

In this case keeping the traffic demand non-uniform we have reduced the capacity of the link (1-2) to observe whether this additional non-symmetry introduced with respect to the capacities in the network will have any effects with respect to demand splitting behavior which we documented above (4.3.2.3) and to see if there is any gain in link utilization with LISP-TE II.

As you can see from the below Table 13; it is evident that with respect to LISP-TE II we can achieve better link utilization as we reduce the capacity of the link (1-2) compared to Base-TE case. Furthermore, we see more traffic being proportioned (for demand volume between router 3 and 2) in addition to the traffic splitting between routers 4 and 2.

Table 13: Non-Uniform Demand, Capacity adjusted on link 1-2

(a) Base Case

Capacity	r
300	0.72
200	0.84
100	1.01

(b) LISP-TE II

Capacity	r	α_{12}	α_{12}^5	α_{32}	α_{32}^5	α_{42}	α_{42}^5
300	0.64	1	0	1	0	0.25	0.74
200	0.64	1	0	1	0	0.25	0.74
100	0.68	1	0	0.91	0.08	0.16	0.83

Thus, LISP-TE II shows more benefits compared to Base-TE in minimizing the maximum link utilization with non-symmetric network case.

4.3.3 Moderate-Size Topologies Considered

We have considered three moderate-size topologies namely Internet2; which is a network research bed hosted/maintained by Indiana University offering full range of network services for research and education purposes [15], AboveNet and Exodus; which are actual ISP topologies drawn from Rocketfuel ISP topology mapping engine [13]. Note that in all the topology each group are selected based on RLOCs proximity to each other.

Before we apply our LISP-TE model to the above mentioned topologies, we first clarify different information we have presented in the Tables 16, 19 and 22; below for their respective topologies.

- Under the capacity column; 10,000 refers to capacity fixed for all the links in the given network; 5,000 (RLOCs) indicates that only the link capacity from any router directly connected to the RLOC routers in all the groups (G^*) is

reduced to 5,000 while all other link capacities in the network remains unaltered.

- Under the demand column;
  - *Uniform* means that the demand volume between all the nodes/routers in a given network is uniformly fixed.
  - *1.5 times (RLOCs)* means that only the demand volume to the destination RLOC routers in groups (i.e.G^*) are increased by 1.5 times from the original demands to those routers.
  - Similarly, *2 times (RLOCs)* means that only the demand volume to the destination RLOC routers in groups (i.e.G^*) are increased by 1.5 times from the original demands to those routers.
  - *Generated* refers to the case where the demands between the nodes/routers are generated using the demand generation model in [15].
    - Under *Total demands column*; "+ n" specifies that for the total demand volume in the network (n) units of additional demand volume have been added due to the considered scenario.
    - Under RLCO demands column, "+ n" specifies that for the total demand volume for RLOCs in the network (n) units of additional demand volume have been added due to the considered scenario.
    - Finally, with separate columns for Total demands and RLOCs demands we have tried to provide a perspective of difference

in demand volume in the entire network compared to the demand volume for the RLOC routers in the same network.

4.3.3.1 Internet2 Topology

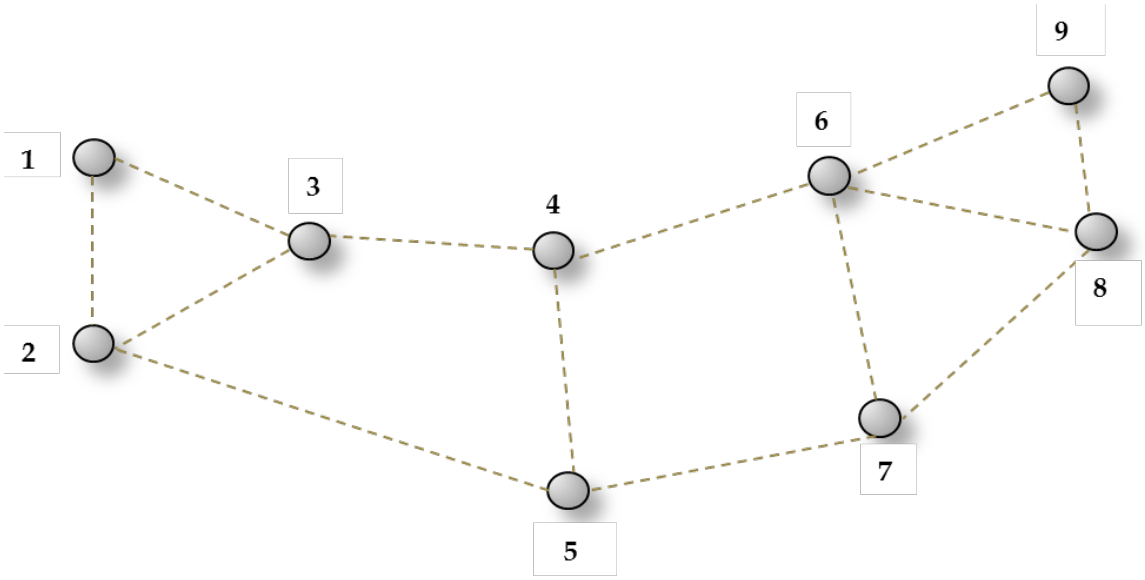

Figure 25: Internet2 Topology

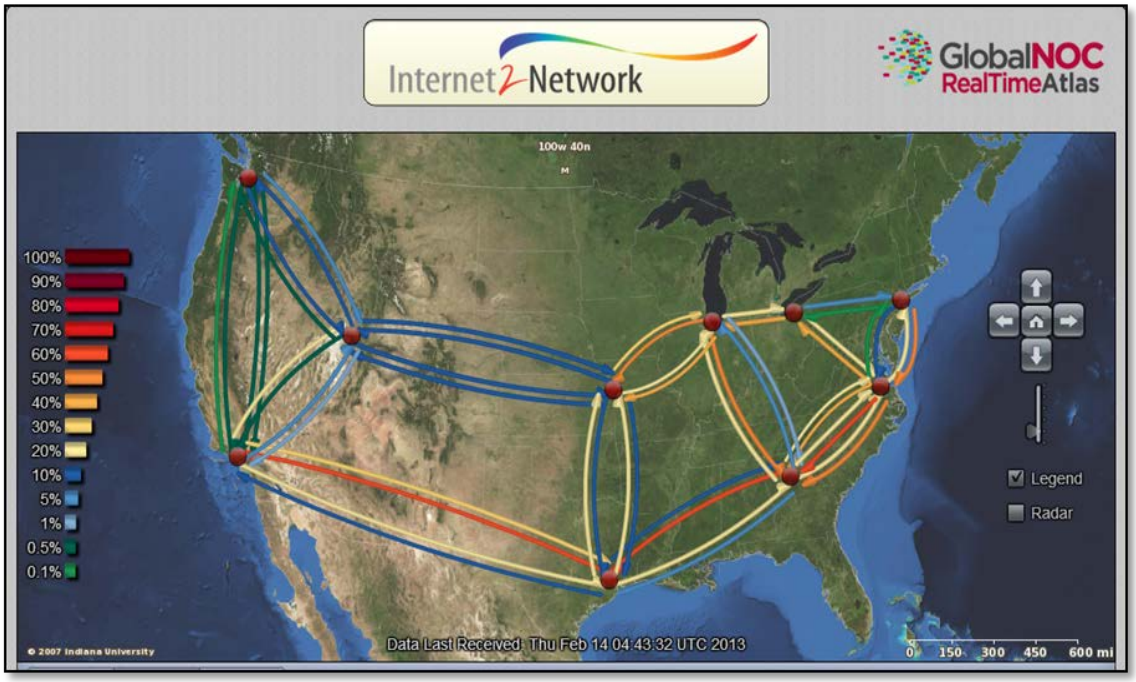

Figure 26: Pictorial Representation of Internet2 Topology

Table 14: Nodes, Links and Co-ordinates in Internet2 Topology

Nodes (Routers)	Links	Co-ordinates
1. Seattle, WA	2, 3	47.609722, -122.333056
2. Los Angeles, CA	3, 5	34.05, -118.25
3. Salt Lake City, UT	4	40.75, -111.88333
4. Kansas City, MO	5, 6	39.099722, -94.578333
5. Houston, TX	7	29.762778, -95.38305
6. Chicago, IL	7, 8, 9	41.881944, -87.627778
7. Atlanta, GA	8	33.755, -84.39
8. Mclean, VA	9	38.934167, -77.1775
9. New York, NY		40.664167, -73.938611

Table 15: RLOCs Group in Internet2 Topology

Groups	Nodes
G^6	6,7,8

Table 16: Results for Internet2 Topology

Capacity	Demands	Total Demands	RLOC Demands	*r*		
				(Base-TE)	(LISP-TE II)	(LISP-TE III)
10000	Uniform(500)	33000	9000	0.5000	0.5000	0.5000
	750 (RLOCs)	+ 4500	+ 4500	0.6875	0.6875	0.5000
	1000 (RLOCs)	+ 9000	+ 9000	0.8750	0.8750	0.5000
5000 (RLOCs)	Uniform(500)	33000	9000	1.0000	1.0000	0.6250
	750 (RLOCs)	+ 4500	+ 4500	1.3750	1.3750	0.8125
	1000 (RLOCs)	+ 9000	+ 9000	1.7500	1.7500	1.0000
10000	Non-Uniform	14878	4046	0.2344	0.2344	0.2344
	1.5 times (RLOCs)	+ 2017	+ 2017	0.3190	0.3190	0.2344
	2 times (RLOCs)	+ 4046	+ 4046	0.4040	0.4040	0.2346
5000 (RLOCs)	Non-Uniform	14878	4046	0.4689	0.4689	0.2994
	1.5 times (RLOCs)	+ 2017	+ 2017	0.6380	0.6380	0.3841
	2 times (RLOCs)	+ 4046	+ 4046	0.8081	0.8081	0.4692

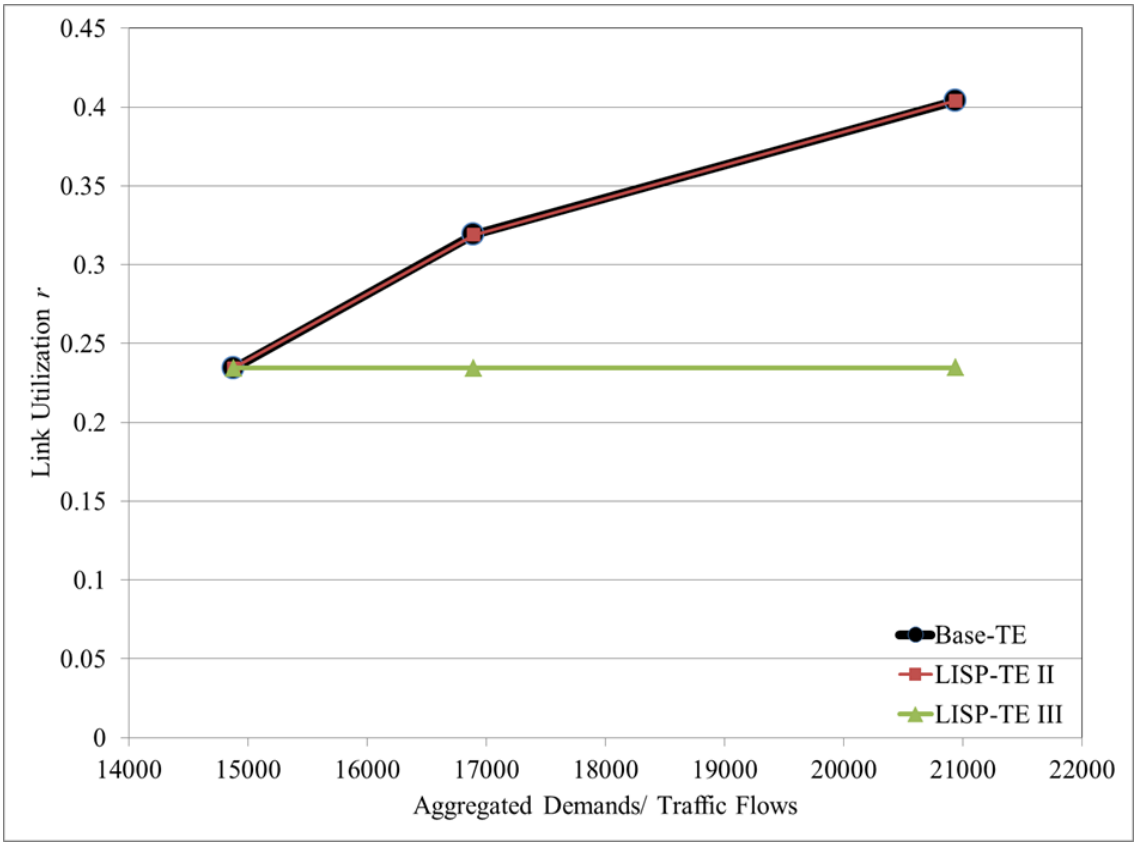

Figure 27: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for Internet2 with Capacity of 10,000 and Non-Uniform Demands

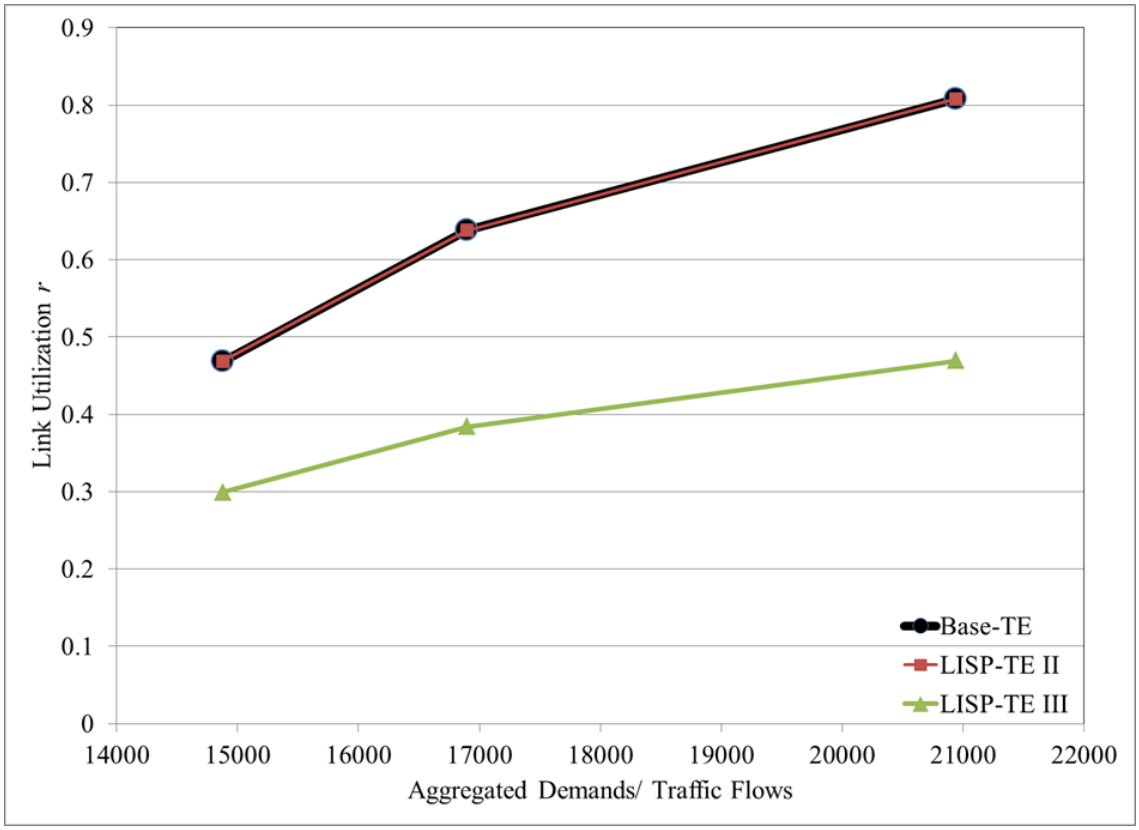

Figure 28: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for Internet2 with Capacity of 5,000 (RLOCs) and Non-Uniform Demands

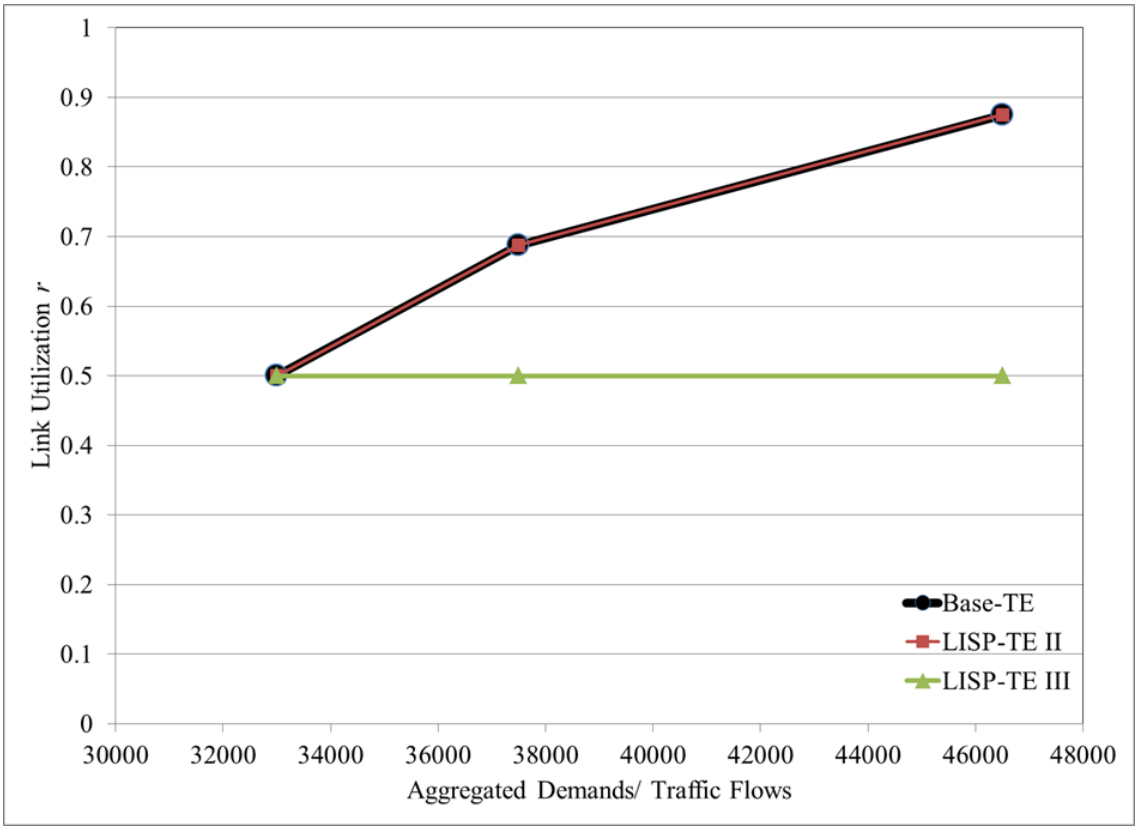

Figure 29: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for Internet2 with Capacity of 10,000 and Uniform Demands

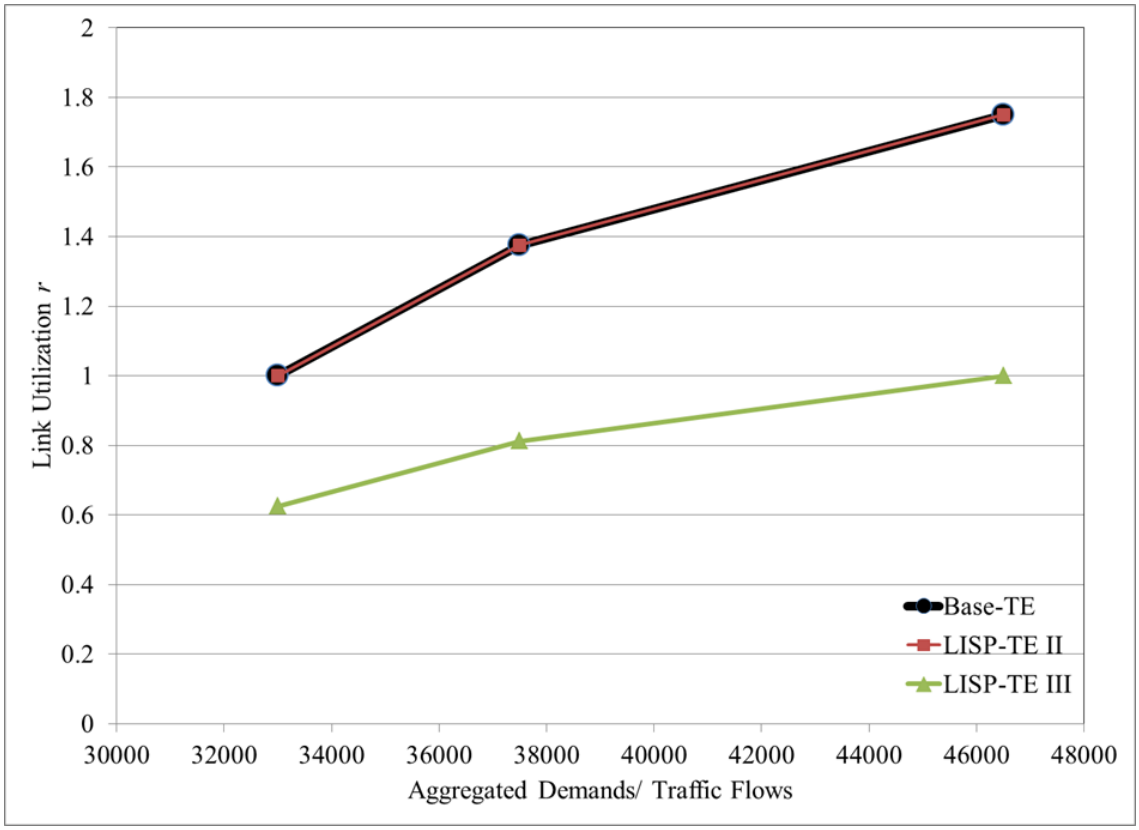

Figure 30: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for Internet2 with Capacity of 5,000 (RLOCs) and Uniform Demands

Optimal r for Internet2 Topology

From the Table 16; we can observe that the optimal *r* remains the same with respect to uniform load case with uniform link capacities throughout the network i.e. there are no benefits of LISP traffic engineering (both LISP-TE II and LISP-TE III) compared to Base-TE in such network conditions. But, when we increase the demands to RLOCs both in generated and in uniform case LISP-TE III gives us better (minimum) *r* compared to the Base-TE as more and more traffic is proportioned to all the peers in the group with respect LISP-TE III. Similarly, when we reduce the capacities to links connecting to RLOCs both in uniform and generated case we see better *r* value compared to Base-TE. The same is depicted in the graphs from Fig. 27 to Fig. 30.

4.3.3.2 AboveNet Topology

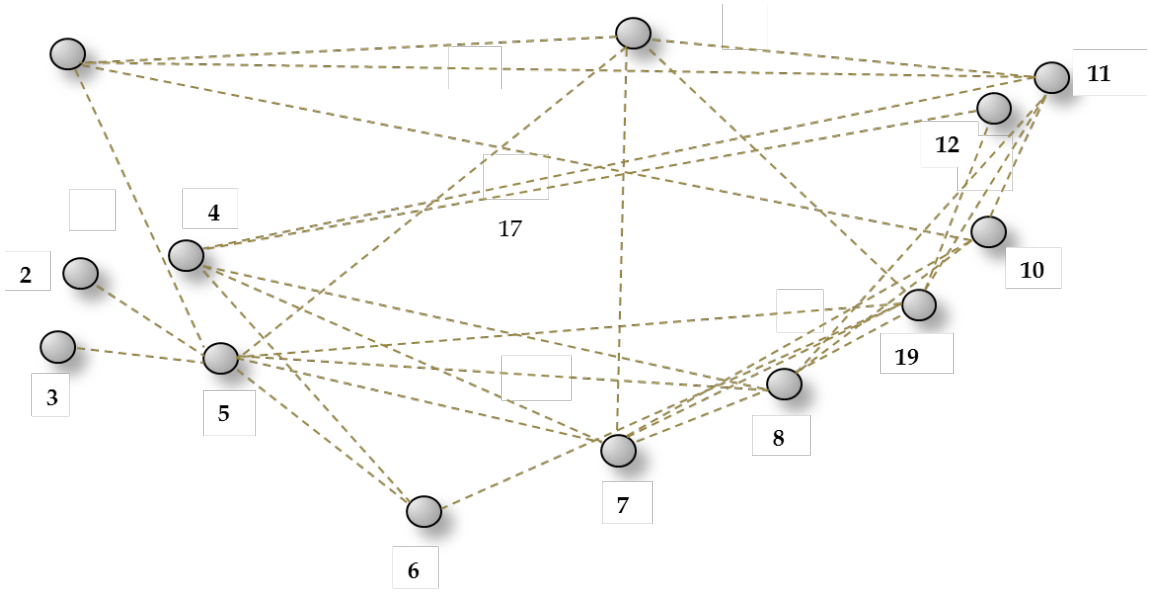

Figure 31: AboveNet Topology

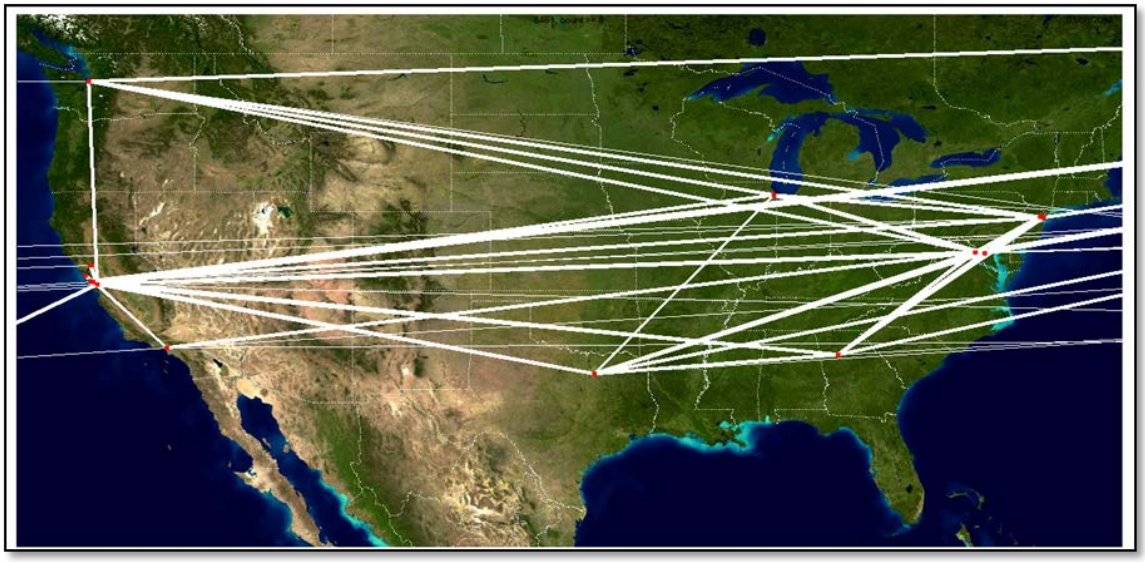

Figure 32: Pictorial Representation of AboveNet Topology

Table 17: Nodes, Links and Co-ordinates in AboveNet Topology

Nodes (Routers)	Links	Co-ordinates
1. Seattle, WA		47.478333, -122.275556
2. Napa, CA	5	38.304722, -122.298889
3. San Francisco, CA	5	37.783333, -122.416667
4. Palo Alto, CA	6, 7, 8	37.429167, -122.138056
5. San Jose, CA	1, 7, 8, 13	37.335278, -121.891944
6. Los Angeles, CA	5	34.05, -118.25
7. Dallas, TX	9, 10, 13	32.782778, -96.803889
8. Atlanta, GA	7, 12	33.755, -84.39
9. IAD	5, 6, 8, 12, 13	38.944444, -77.455833
10. Washington, DC	1	38.895111, -77.036667
11. New York, NY	1, 4, 8, 9, 10, 13	40.664167, -73.938611
12. Newark, NY	4	40.72422, -74.172574
13. Chicago, IL	1	41.881944, -87.627778

Table 18: RLOCs Groups in AboveNet Topology

Groups	Nodes
G^{2}	2, 3, 5, 6
G^{11}	11, 12

Table 19: Results for AboveNet Topology

Capacity	Demands	Total Demands	RLOC Demands	*R*		
				(Base-TE)	(LISP-TE II)	(LISP-TE III)
10000	Uniform(200)	28400	11600	0.1800	0.1800	0.1800
	350 (RLOCs)	+ 5800	+ 5800	0.2700	0.2700	0.2000
	400 (RLOCs)	+ 11600	+ 11600	0.3600	0.3600	0.2200
5000 (RLOCs)	Uniform(200)	28400	11600	0.3600	0.3600	0.3600
	350 (RLOCs)	+ 5800	+ 5800	0.5400	0.5400	0.4000
	400 (RLOCs)	+ 11600	+ 11600	0.7200	0.7200	0.4400
10000	Non-Uniform	34042	14407	0.2217	0.2217	0.2217
	1.5 times (RLOCs)	+ 7191	+ 7191	0.3323	0.3152	0.2470
	2 times (RLOCs)	+ 14407	+ 14407	0.4434	0.4206	0.2724
5000 (RLOCs)	Non-Uniform	34042	14407	0.4434	0.4434	0.4434
	1.5 times (RLOCs)	+ 7191	+ 7191	0.6646	0.6304	0.4940
	2 times (RLOCs)	+ 14407	+ 14407	0.8868	0.8412	0.5448

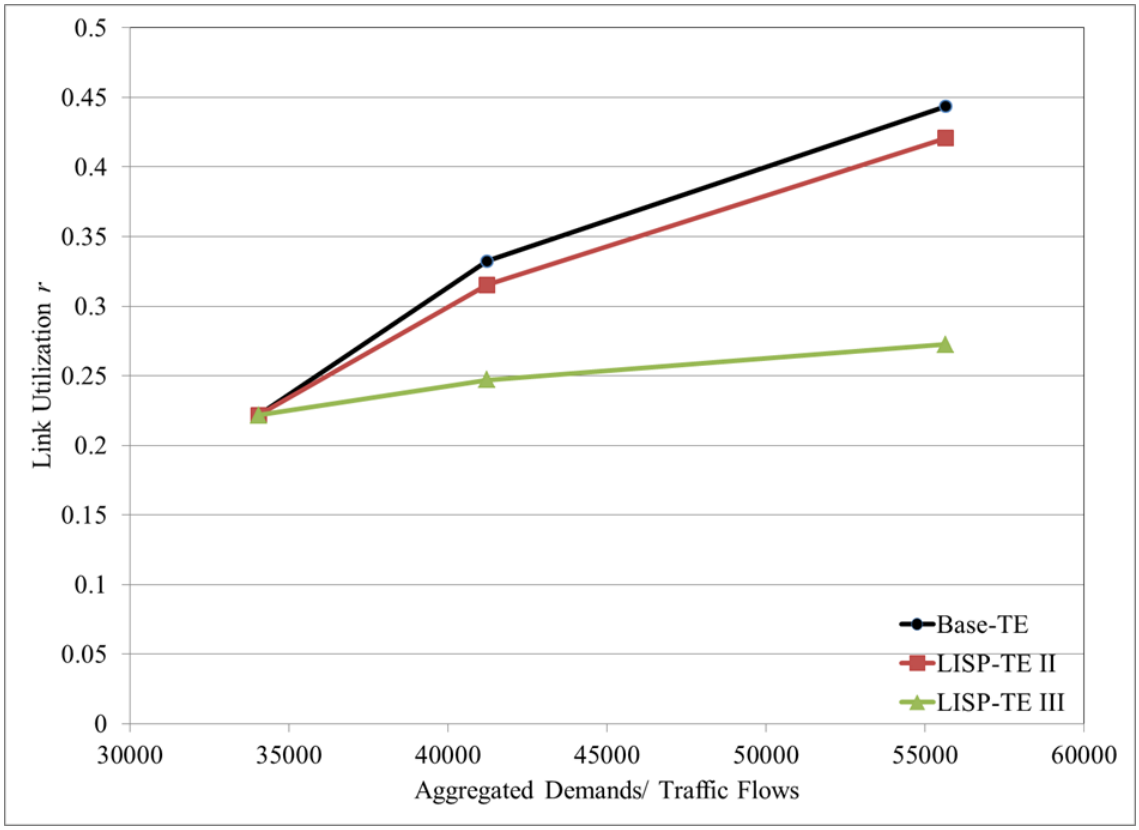

Figure 35: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for AboveNet with Capacity of 10,000 Non-Uniform Demands

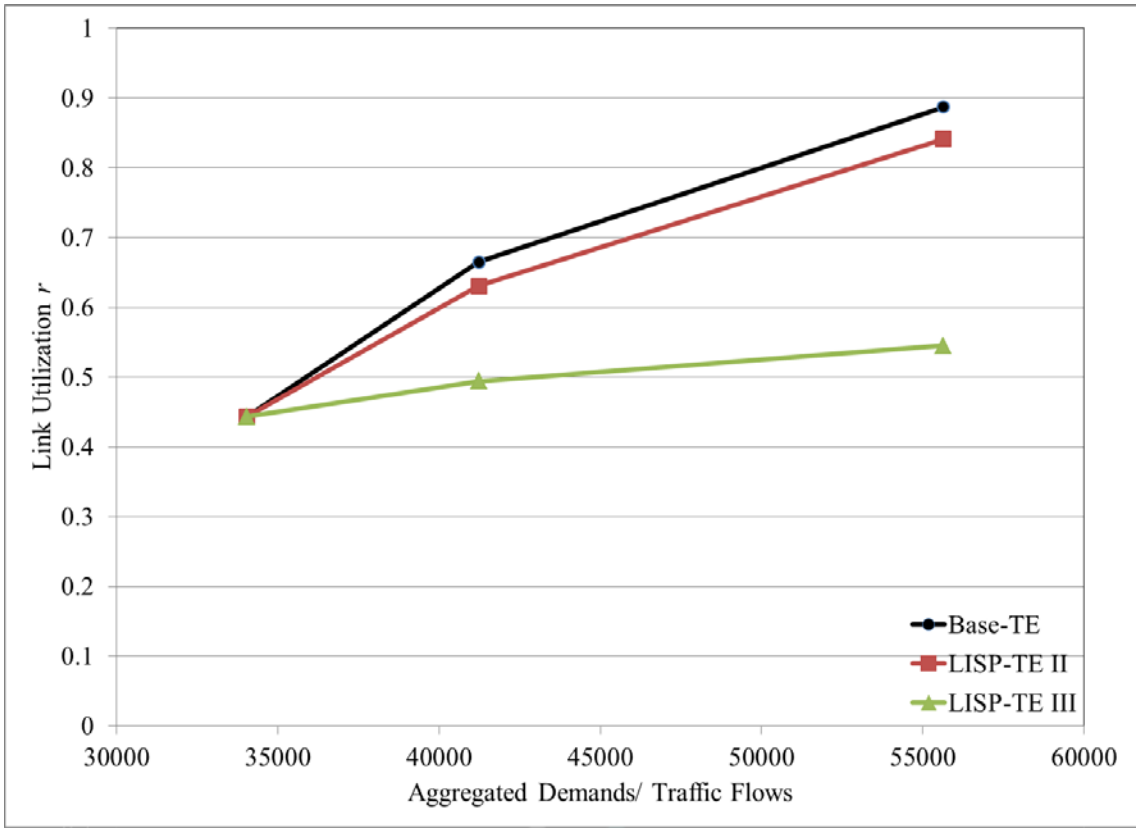

Figure 34: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for AboveNet with Capacity of 5,000 (RLOCs) and Non-Uniform Demands

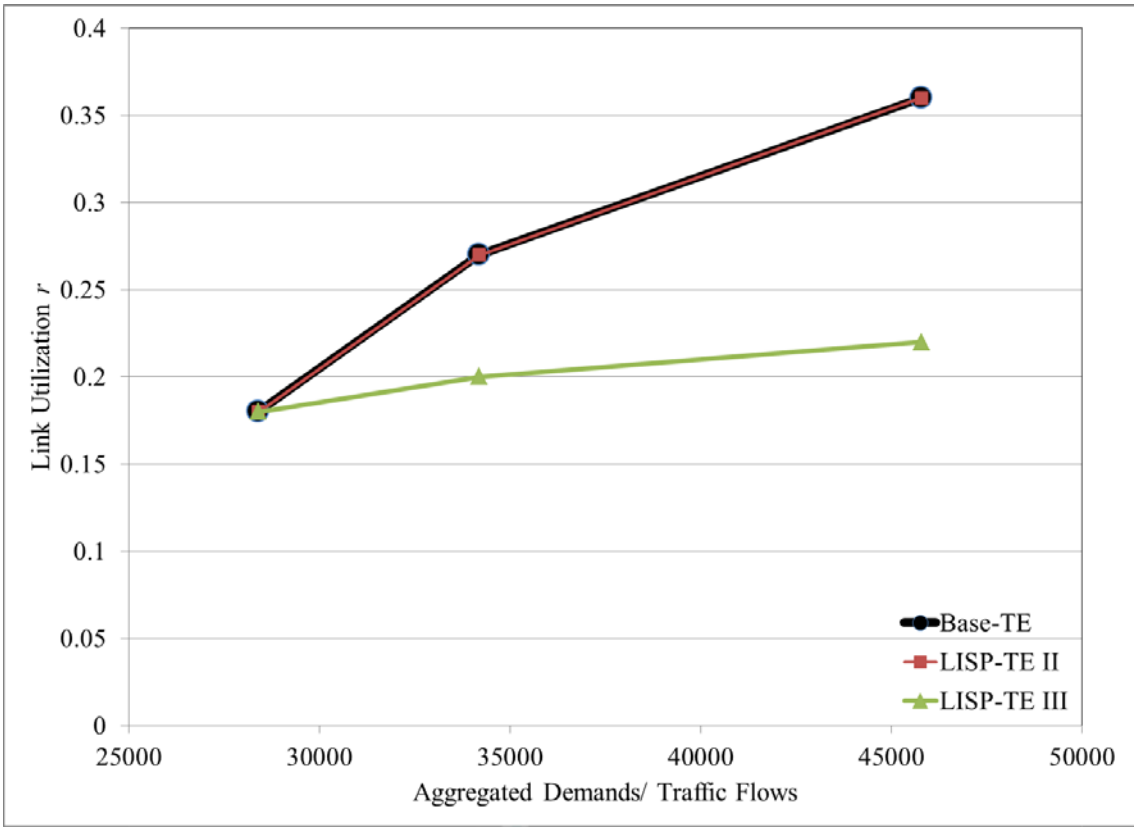

Figure 35: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for AboveNet with Capacity of 10,000 Uniform Demands

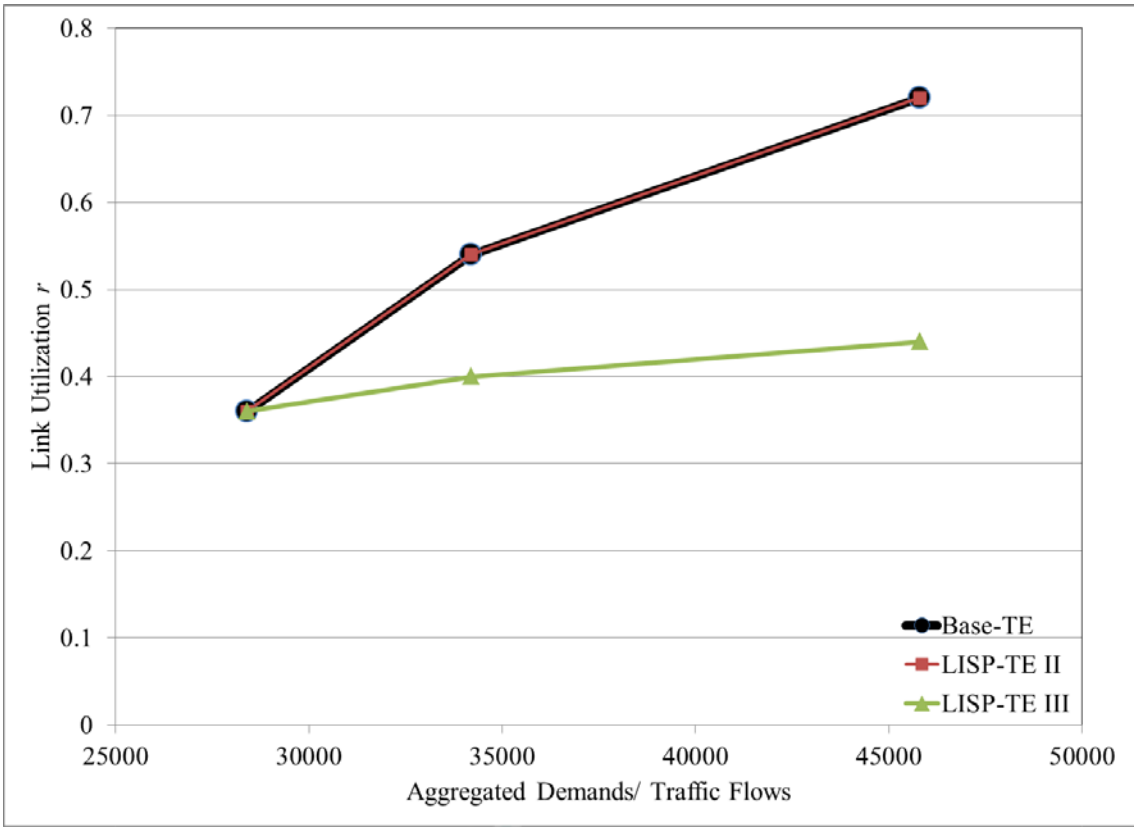

Figure 36: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for AboveNet with Capacity of 5,000 (RLOCs) and Uniform Demands

Optimal *r* for AboveNet Topology

Similar to Internet2 topology, the optimal *r* remains the same with respect to uniform load case with uniform link capacities throughout the network i.e. there are no benefits of LISP traffic engineering (both LISP-TE II and LISP-TE III) compared to Base-TE. But, unlike Internet2 in case of generated demand when we increase the demands to RLOCs as well as decreasing the capacities to the links connecting to RLOCs we see the benefits of LISP traffic engineering (both LISP-TE II and LISP III) with respect to the optimal value of *r* compared to the Base-TE. The same is depicted in the graphs from Fig. 33 to Fig. 36.

4.3.3.3 Exodus Topology

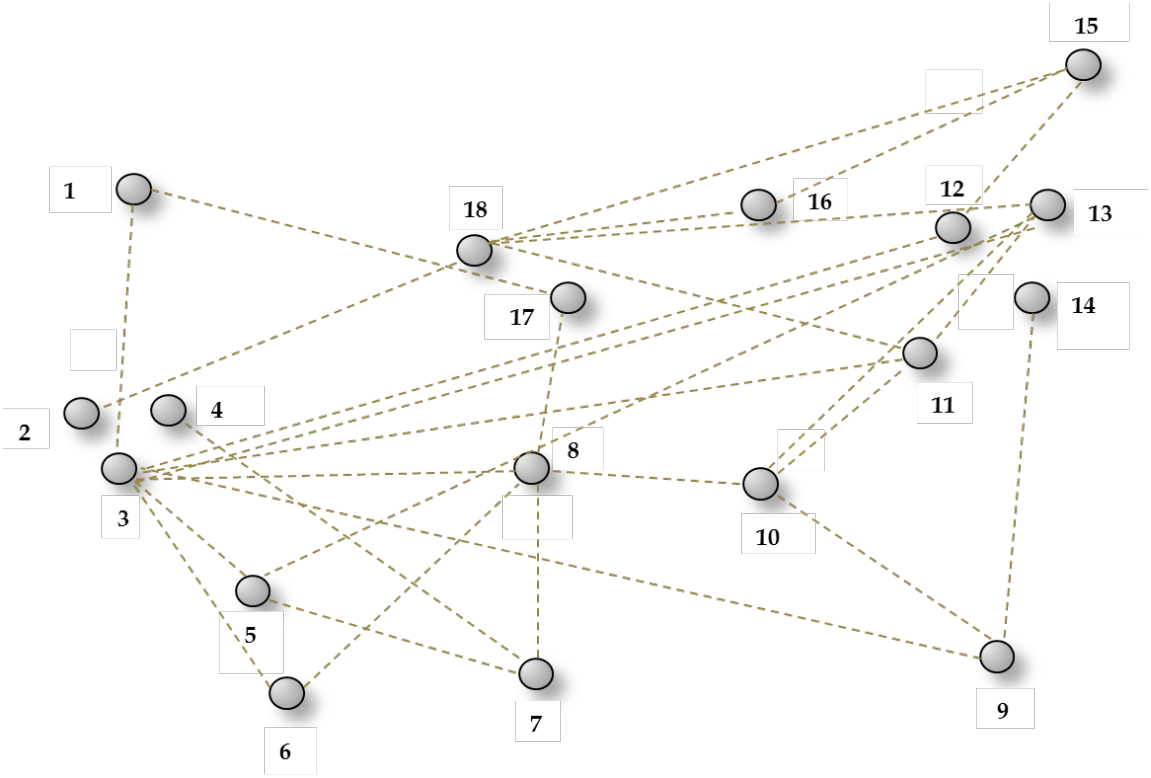

Figure 37: Exodus Topology

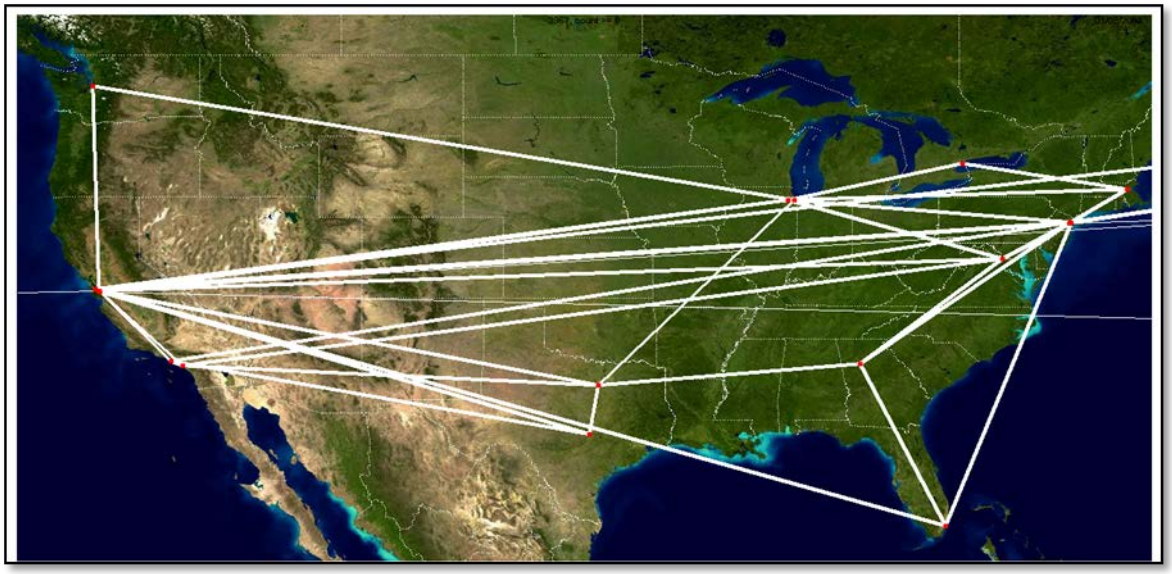

Figure 38: Pictorial Representation of Exodus Topology

Table 20: Nodes, Links and Co-ordinates in Exodus Topology

Nodes (Routers)	Links	Co-ordinates
1. Tukwila, WA	3, 17	47.478333, -122.275556
2. Palo Alto, CA	18, 13	37.429167, -122.138056
3. Santa Clara, CA	12, 9, 5	37.354444, -121.969167
4. San Jose, CA	7	37.335278, -121.891944
5. El Segundo, CA	7	33.921389, -118.406111
6. Irvine, CA	3, 11	33.684167, -117.7925
7. Austin, TX		30.25, -97.75
8. Fort Worth, TX	3, 6, 7, 17, 10	32.757358, -97.333181
9. Miami, FL	10	25.787778, -80.224167
10. Atlanta, GA		33.755, -84.39
11. Herndon, VA	3, 10, 18	38.971389, -77.388611
12. Jersey City, NJ		40.711417, -74.06476
13. Weehawken, NY	3, 5, 10, 11, 18	40.768903, -74.015427
14. New York, NY	9	40.664167, -73.938611
15. Waltham, MA	12, 18	42.376389, -71.236111
16. Toronto, CA	15, 18	43.716589, -79.340686
17. Chicago, IL		41.881944, -87.627778
18. Oak Brook, IL		41.84, -87.953056

Table 21: RLOCs Groups in Exodus Topology

Groups	Nodes
G^3	3, 5, 6
G^7	7, 8, 10
G^{12}	12, 13

Table 22: Results for Exodus Topology

Capacity	Demands	Total Demands	RLOC Demands	*r*		
				(Base-TE)	(LISP-TE II)	(LISP-TE III)
10000	Uniform(200)	58400	24400	0.3400	0.3400	0.3400
	350 (RLOCs)	+ 12200	+ 12200	0.4200	0.4200	0.3400
	400 (RLOCs)	+ 24400	+ 24400	0.5000	0.5000	0.3800
5000 (RLOCs)	Uniform(200)	58400	24400	0.6800	0.6800	0.5600
	350 (RLOCs)	+ 12200	+ 12200	0.8400	0.8400	0.6600
	400 (RLOCs)	+ 24400	+ 24400	1.0000	1.0000	0.7600
10000	Generated	62128	26273	0.3978	0.3978	0.3978
	1.5 times (RLOCs)	+ 13108	+ 13108	0.4930	0.4930	0.3978
	2 times (RLOCs)	+ 26273	+ 26273	0.5981	0.5981	0.4607
5000 (RLOCs)	Generated	62128	26273	0.7956	0.7956	0.6567
	1.5 times (RLOCs)	+ 13108	+ 13108	0.9860	0.9860	0.7778
	2 times (RLOCs)	+ 26273	+ 26273	1.1772	1.1772	0.8994

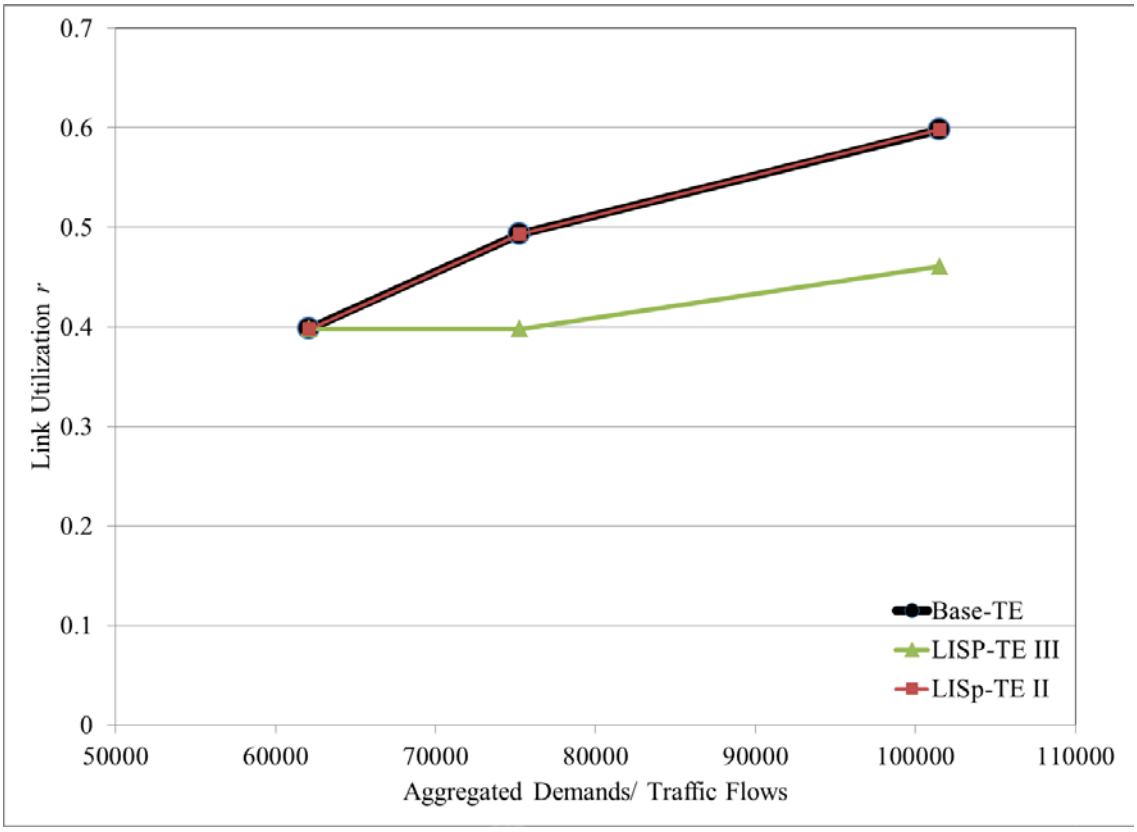

Figure 39: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for Exodus with Capacity of 10,000 and Non-Uniform Demands

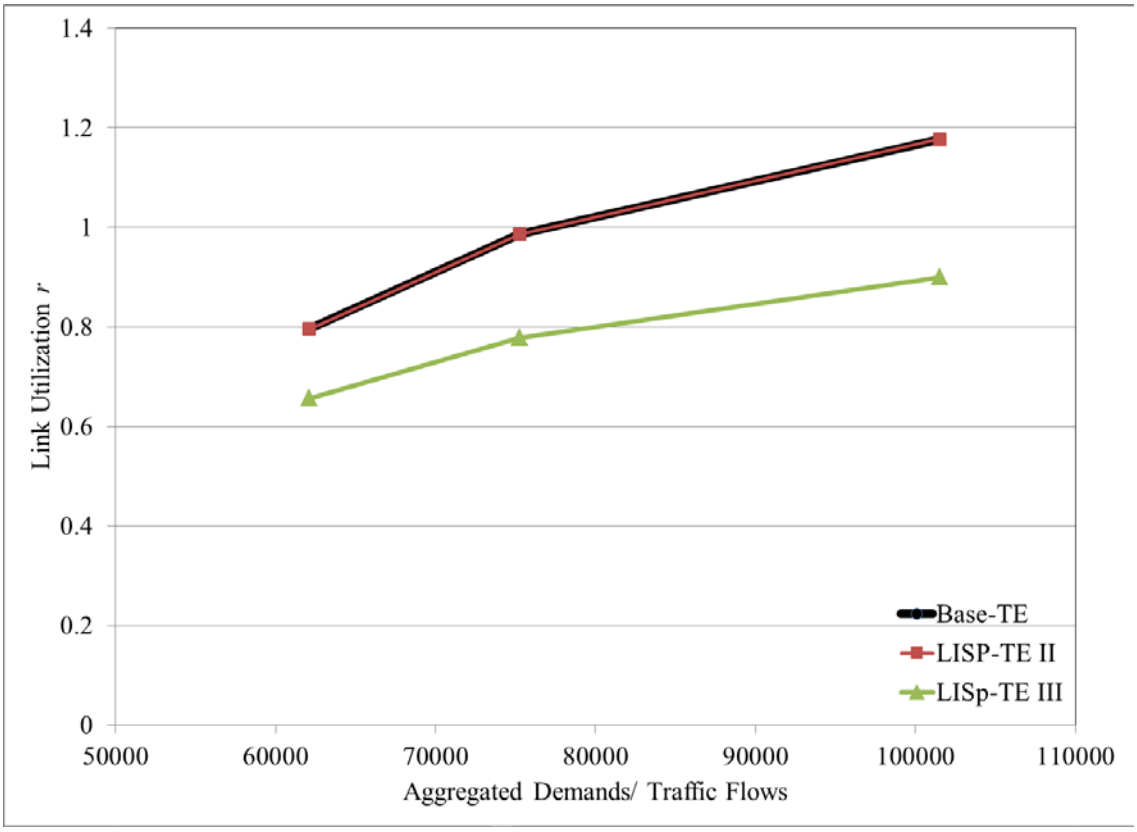

Figure 40: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for Exodus with Capacity of 5,000(RLOCs) and Non-Uniform Demands

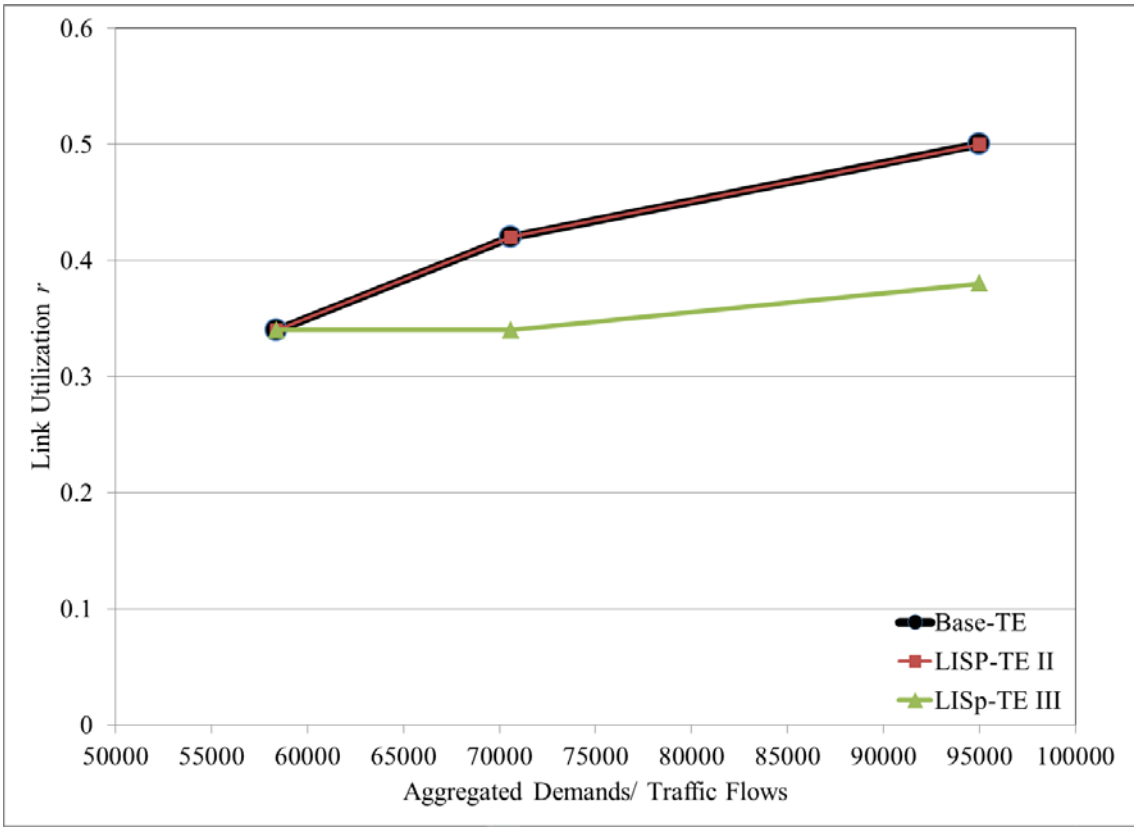

Figure 41: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for Exodus with Capacity of 10,000 and Uniform Demands

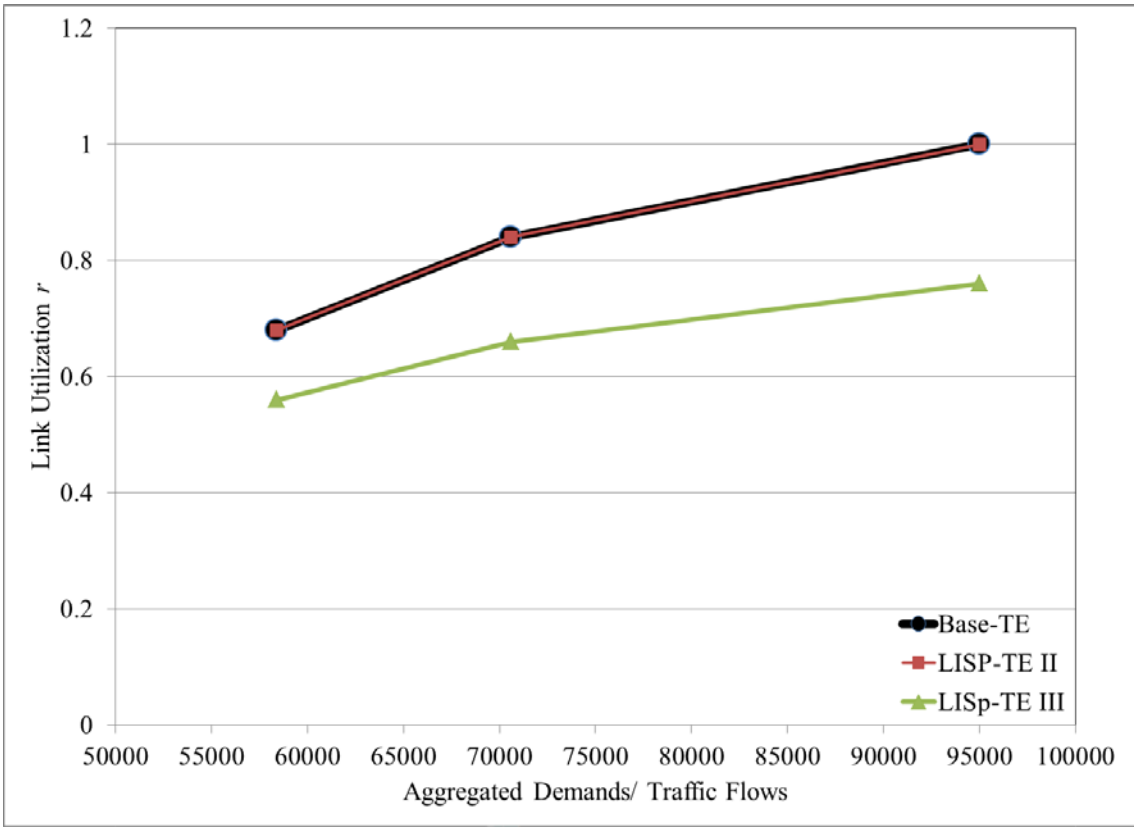

Figure 42: Comparison of *r (*Base-TE v/s LISP-TE II v/s LISP-TE III*)* value for Exodus with Capacity of 5,000(RLOCs) and Uniform Demands

Optimal r for Exodus Topology

Again, similar to both Internet2 and AboveNet topologies, the optimal *r* remains the same with respect to uniform load case with uniform link capacities throughout the network i.e. there are no benefits of LISP traffic engineering (both LISP-TE II and LISP-TE III) compared to Base-TE in such network conditions and when we increase the demands to RLOCs both in generated and in uniform case LISP-TE III gives us better (minimum) *r* compared to the Base-TE as more and more traffic is proportioned to all the peers in the group with respect LISP-TE III. Similarly, when we reduce the capacities to links connecting to RLOCs both in uniform and generated case we see better *r* value compared to Base-TE. Note that for one of the scenario we considered with generated demand case with increased demands to RLOCs and reduced capacities to the links connecting to the RLOCs, optimal *r* becomes infeasible in Base-TE, whereas with LISP-TE III optimal *r* even though high remains less than 1, the same is depicted in graphs from Fig. 39 to Fig. 42. Thus showing the benefits of traffic proportioning in LISP-enabled networks compared to Base-TE where traffic proportioning is not possible because of lack of path diversity.

CHAPTER 5

SUMMARY

In this thesis our main objective is to understand the benefits that a LISP-enabled network where, in such a network, it is possible to proportion the traffic to multiple RLOCs associated to a single destination EIDs; offers with respect to inter-domain traffic engineering compared to the current routing architecture; where, effective traffic engineering is achieved through de-aggregating prefixes because of the rigid rule of "*Advertising the single best path*" that BGP follows.

In order to achieve our goal, we have presented two models for traffic engineering in LISP-enabled network by introducing the concept of "*grouping"* multiple RLOCs with traffic proportioning or load-balancing as the optimization criterion. In the first model called "LISP-TE II" traffic proportioning may be achieved between RLOCs in a group only if the traffic is destined for one of the RLOCs identified as the "*primary*" RLOC for that group. In our second model called "LISP-TE III" traffic proportioning may be achieved between RLOCs in a group if the traffic is destined to any one of the RLOCs in that same group. To evaluate the effectiveness of LISP-enabled traffic engineering over today's traffic engineering practices where traffic proportioning is not available which we call Base-TE, we have applied our formulations to three topologies, namely; Internet2, AboveNet and Exodus where the latter two are current existing ISP topologies in the Internet obtained through Rocketfuel ISP topology engine [15]. Through our study we show that LISP-TE is most effective when the

network load and the capacity are asymmetric/non-uniform, where LISP-TE takes advantage of multiple RLOCs by proportioning traffic to these RLOCs to provide better network utilization compared to the Base-TE where traffic proportioning is not available. On the other hand, when the network load and the capacity is symmetric/uniform, we see very little gain with LISP-TE compared to Base-TE with respect to optimizing network link utilization.

CHAPTER 6

FUTURE WORK

In our thesis work, the objective function of our optimization models is minimizing the maximum link utilization, in our future work; we would like to evaluate our models with different optimization criterion like “Minimum Cost Routing” and “Minimization of Delay”. Also, we would like to model our formulations in “Multi-time Periods”.

In our formulation, the notion of *grouping* multiple RLOCs is introduced to proportion the traffic in a LISP-enabled network, where RLOCs are grouped based on their geographical proximity, in our future studies we would like to investigate different criterion apart from geographical proximity for *grouping* RLOCs like number of EIDs associated to a RLOC or whether an RLOC is a customer edge or provider edge router etc. Furthermore, to evaluate our optimization models we have made some assumptions like there is no intra-group traffic between RLOCs, in our future studies, we are looking into situation where we can relax this assumption so that we can evaluate and compare our models with different sets of RLOCs group to see which combination provides best results.

REFERENCES

[1] Meyer, D., Zhang, L. and K. Fall. *Routing and Addressing*. Internet Architecture Board, Amsterdam, 2007

[2] Farinacci, D., Fuller, V., Meyer, D., and Lewis, D. *The Locator/ID Separation Protocol (LISP)*. Available from http://tools.ietf.org/html/rfc6830 (2013).

[3] Lewis, D., Meyer, D., Farinacci, D., and Fuller, V. *Interworking LISP with IPv4 and IPv6*. Cisco Systems, Inc., 2010.

[4] Farinacci, D., Fuller, V, Meyer, D., and Lewis, D. *LISP Alternative Topology (LISP-ALT)*. Cisco, 2010.

[5] Medhi, D. and Ramasamy, K. *Network Routing: Algorithms, Protocols and Architectures*. Elsevier Inc., San Francisco, 2007.

[6] Quitin, B., Tandel, S., Uhli, S., and Bonaventure, O. Interdomain Traffic Engineering with Reditribution Communities. *Computer Communications*, 27 (March 2004), 355-363.

[7] Quoitin, B., Pelsser, C., Swinnen, L., Bonaventure, O., and Uhlig, S. *Interdomain Traffic Engineering with BGP*. University of Namur, Belgium, 2003.

[8] Quoitin, B., Iannone, L., De Launois, C., and Banoaventure, O. Evaluating the benefits of the locator/identifier separation. In *2nd ACM/IEEE international workshop* (New York 2007), ACM.

[9] Narten, T. *Scalability of Internet Routing*. IBM, New York, 2010.

[10] Saucez, D., Donnet, B., Iannone, L., and Bonaventure, O. Interdomain Traffic Engineering in a locator/identifier seperation context. In *Internet Network Management Workshop* (2008), IEEE, 1-6.

[11] Pioro, M. and Medhi, D. *Routing, Flow, and Capacity Design in Communication and Computer Networks*. Elsevier Inc, San Francisco, 2004.

[12] Fortz, B. and Thorup, M. Internet Traffic Engineering by optimizing ospf weights in INFOCOM 2000. In *Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies* (2002), IEEE, 519-528.

[13] Spring, N., Mahajan, R., and Wetherall, D. Measuring ISP topologies with Rocketfuel. *Transactions on Networking*, 12, 1 (February 2004), 2-16.

[14] Dasgupta, S., Sridhar, R., and Secci, S. Medhi, D. *Traffic Engineering with Multiple RLOCs in Cooperative LISP-enabled Networks*. 2013. Unpublished Report.

[15] GlobalNOC. http://atlas.grnoc.iu.edu/atlas.cgi?map_name=Internet2%20IP%20Layer

[16] Jain, V. *A Study of Locator and ID Separation Protocol*. MS Thesis, University of Missouri Kansas City, School of Computing and Engineering, Kansas City, 2010.

VITA

Raghunandan Sridhar was born in December 18, 1985 in Mysore City, Karnataka, India. He was educated in Savidya Patashalla and graduated with distinction from High School in 2003. After graduating, he attended P.S.E Engineering College, which is a part of prestigious Visveshvariah Technological University (VTU) and graduated in July 2007 with a Bachelor of Engineering degree in Computer Science.

After obtaining his undergraduate degree, he worked as an Associate Software Engineer from August 2007 to July 2009 at Accenture, India. He resigned from Accenture to pursue higher education and joined Master's program at University of Missouri-Kansas City, Missouri, USA. From December 2011, he has been a member of Network Research Laboratory (NetRel) at School of Computing and Engineering. In March 2012, he joined Gambit Communications, Inc as a full-time employee where he worked in the capacity as a Network Software Engineer related to his research work and his major, Computer Networking. After working 7 months at Gambit communications he resigned to pursue a new and more challenging position with ADTRAN, Inc. Currently, he is working full-time as a Software Engineer (Layer 2 Technologies, related to his thesis and major, Computer Networking) at ADTRAN, Inc.

Upon completion of his Master's degree requirement, he plans to continue his employment at ADTRAN, Inc and one day he hopes to be a major contributor in the field of Computer Networking.