On the Cost of Supporting Multihoming and Mobility

Vatche Ishakian   Joseph Akinwumi   Ibrahim Matta
Computer Science
Boston University
Boston, MA, USA
{visahak, akin, matta}@cs.bu.edu

Technical Report BUCS-TR-2009-020
June 2009

ABSTRACT

As the Internet has evolved and grown, an increasing number of nodes (hosts or autonomous systems) have become multihomed, i.e., a node is connected to more than one network. Mobility can be viewed as a special case of multihoming—as a node moves, it unsubscribes from one network and subscribes to another, which is akin to one interface becoming inactive and another active. The current Internet architecture has been facing significant challenges in effectively dealing with multihoming (and consequently mobility). The Recursive INternet Architecture (RINA) [1] was recently proposed as a clean-slate solution to the current problems of the Internet. In this paper, we perform an average-case cost analysis to compare the multihoming / mobility support of RINA, against that of other approaches such as LISP and Mobile-IP. We also validate our analysis using trace-driven simulation.

1. INTRODUCTION

Support for multihoming and mobility was not a primary goal in the original design of the Internet. As a result, the Internet's naming and addressing architecture is incomplete. Specifically, the address of a multihomed host specifies a particular interface (connection), rather than the node itself. Because routing is done based on this interface (IP) address, if this active interface goes down, it's costly to switch to another operational interface.

There have been several attempts to fix this addressing problem, including the Location ID Separation Protocol (LISP)—currently being tested at Cisco [4, 8]—and Mobile-IP [10]. The basic idea behind LISP is to assign the multihomed node a provider-independent (location-independent) identifier (ID). A border router maps a destination ID to the node's location, which is the address of another border router that is known to have a path to the node. Routing is then done from the source's border router to the destination's border router. If the latter (node's location) changes due to path failure or mobility, it becomes costly to propagate that change over the whole Internet (to all possible source border routers).

Mobile-IP (MIP) allows a mobile host to seamlessly move from its home domain to a foreign location without losing connectivity. This is done by having a foreign agent update the location of the mobile node at its home agent. Since mobility is a special (dynamic) form of multihoming, MIP can also be used to handle a change in the active interface (due to failure or rerouting) leading to a multihomed node, where a home agent directs traffic to the currently active (operational or "better") interface. However, this location update can be costly since it needs to propagate from the foreign agent to the home agent.

Note that both LISP and Mobile-IP (and combination thereof) help reduce the size of the routing tables at the core of the Internet, since several IDs can map to one location and hence be represented by one routing entry. Further elaboration on the benefits of LISP can be found in [11].

RINA [1] is a recently proposed Recursive INternet Architecture. It uses the concept of Distributed IPC Facility (DIF) to divide communication processes into manageable scopes across network subsystems, which results in a reduced routing table size per DIF. RINA routes hop-by-hop based on the destination's node address, not its interface. At each hop, the next-hop node address is mapped to the (currently operational) interface to that next-hop node. This late binding of a node's address to its interface (path) allows RINA to effectively deal with interface changes due to multihoming or mo-

Boston University, Computer Science, June 2009.

bility. The cost of such late binding is relatively small since its scope is local to the routing "hop" that traverses the underlying DIF. By recursing the DIF structure to make the DIF scopes small enough, the cost of such late bindings (location updates) can be made arbitrarily small.

1.1 Our Contribution

We present a cost model to quantitatively assess the effectiveness of LISP, MIP, and RINA, in supporting multihoming / mobility. To the best of our knowledge, this paper presents a first cost comparison of these approaches. Our definition of "cost" captures both the average number of packets generated by a source node to a (multihomed or mobile) destination node, as well as the average path length from the source to the destination (as indication of delays or bandwidth usage). In our model, we compute the overall average cost for a single interface change experienced by the multihomed or mobile destination node. We also validate our analytical model using trace-driven simulation.

1.2 Organization of the Paper

The rest of the paper is organized as follows. Section 2 reviews MIP, LISP, and RINA. We present our general cost model in Section 3, and then we instantiate it for the various approaches. Section 4 presents numerical results for grid topologies as well as for a typical Internet topology. Section 5 presents the cost of supporting multihoming using real packet traces from CAIDA [12]. Section 6 concludes the paper.

2. BACKGROUND

This section provides a basic background on the various architectures we study, namely MIP, LISP, and RINA—for more details, we refer the reader to references herein.

2.1 Mobile-IP

Mobile-IP (MIP) [10] has been mainly standardized to deal with the mobility of nodes. As mentioned earlier, since mobility is merely a (dynamic) form of multihoming, the MIP concept can also be used to deal with interface (path) change to a multihomed node.

In MIP, two basic mechanisms are identified: (1) a discovery mechanism, which allows a node to detect its new point-of-attachment, and (2) a registration mechanism, which allows a node to register itself with an agent that represents it at its home network.

Figure 1 shows a source node (SN) sending packets to a destination node (DN) in another Autonomous System (AS). The destination moves to a new AS and acquires a care-of-address at the Foreign Agent (FA). The FA then updates the corresponding Home Agent (HA) with DN's new location.

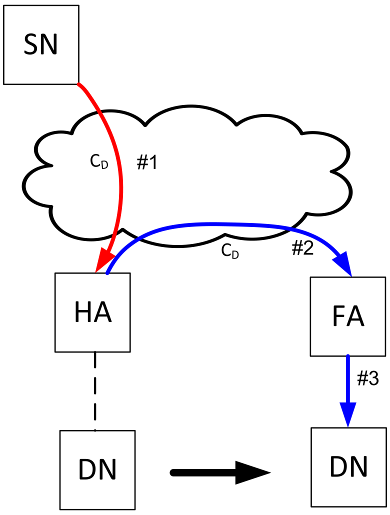

Figure 1: Mobile-IP Protocol

The basic delivery process of data packets from a source node to a destination node is as follows (highlighted as sequence 1–3 in Figure 1):

1. The datagram is delivered to HA via standard routing.
2. The HA intercepts the datagram and tunnels it to the destination's current location (care-of-address).
3. The FA at the current location intercepts the datagram and delivers it to the destination node.

2.2 LISP

The Locator/ID Separation Protocol (LISP), proposed by Farinacci et al. [3], separates the address space into end-systems' identifiers (EID) and routing locators (RLOCs). Border routers act as RLOCs for the end-systems inside their local domain.

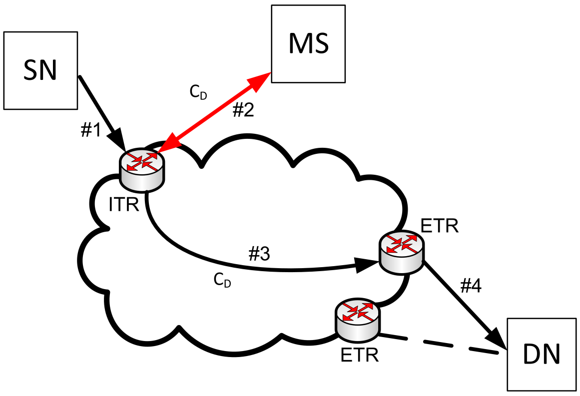

Figure 2: LISP Architecture

The basic delivery process of data packets from a

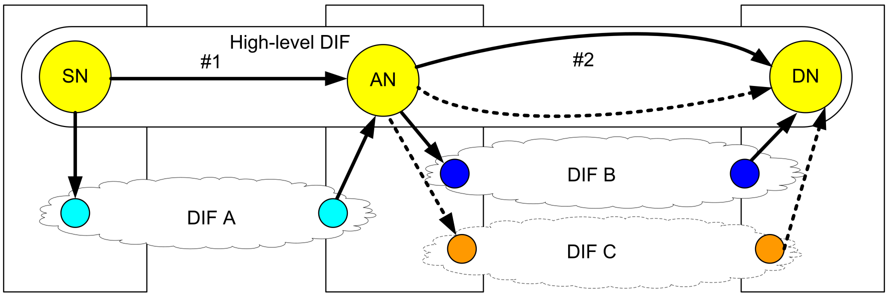

Figure 3: RINA Architecture

source node (SN) to a destination node (DN) is as follows (highlighted as sequence 1–4 in Figure 2):

1. The source forwards the packet to its border router called Ingress Tunnel Router (ITR).
2. The source ITR performs a lookup query for a destination EID-to-RLOC mapping [2].
3. ITR transparently tunnels the data packets to the destination's RLOC referred to as Egress Tunnel Router (ETR).
4. Upon intercepting the packet, the destination's ETR forwards the packet to the destination.

Upon failure of an active interface, a multihomed destination node would send an update to its ETR, which in turn would update the EID-to-RLOC Mapping Server (MS). The sequence of messages is shown in Figure 4.

Different variants of LISP only differ in how the EID-to-RLOC mapping is done [2]. The use of caching for lookup has also been recently explored in [5].

2.3 RINA

In RINA, application processes or services have globally unique names, and networking is viewed as distributed Inter-Process Communication (IPC) [1].

If an application process in RINA needs to communicate with another application process, it requests service from the underlying Distributed IPC Facility (DIF). This DIF maps the destination application name to a node (process) address. A DIF in RINA can (recursively) provide transport services between source and destination application processes, using services of underlying (lower-level) DIFs.

The route to the destination node address (to which the destination application process is connected) is computed as a sequence of intermediate node addresses. At each routing hop, the next-hop node address is in turn mapped (recursively) to a lower-level node address by the underlying DIF. This lower-level node address is viewed as the point-of-attachment (PoA) of the higher-level node. Thus, RINA's addresses are relative. Eventually, the node (process) address maps to a specific path (interface). This late binding to a specific interface (path) makes it easier for RINA to deal with multi-homing (and mobility). If an active interface (path) to a node fails, RINA maps the (next-hop / destination) node address to another operational interface (path). The cost of such interface/location update is small because the update is only local to the routing hop—the next-hop / destination node address is mapped to the lower-level node address that resides within the operational lower-level DIF.

On the contrary, in the current Internet model, the interface address (i.e., IP address) names both the node itself and the interface (path) to that node—this static binding makes multi-homing (and mobility) difficult to manage.

Figure 3 shows a source process (SN) sending packets to a destination process (DN) using the services of the underlying DIFs.[1] SN and DN form a (high-level) DIF with an intermediate process, which we call Ancestor Node (AN), such that AN is connected to the destination DN using two separate interfaces over two different underlying DIFs. This 3-node DIF can be thought of as an "overlay" to which SN, DN, and AN had subscribed. When a packet reaches the Ancestor Node, AN forwards it based on the current best / operational interface leading to the destination DN.

It is important to highlight the difference between how BGP and RINA handle route / interface failures. In BGP, even if there is a specific path failure to a specific prefix (node), BGP would still broadcast a path to the destination since it relies on advertising reachability to aggregate destination prefixes. On the other hand, RINA would handle such failures using hop-by-hop routing within the DIF of the destination process. In Figure 3, if the (solid) overlay link AN–DN that uses the underlying DIF B goes down, node AN would locally adapt and start using the (dotted) overlay link AN–DN that uses the underlying DIF C. Thus, RINA provides finer grained control over routing to multihomed desti-

---

[1] Note that in RINA, a single system may have multiple processes which are members of different DIFs at different levels [1].

nations.

3. COST MODEL

In this section we study the average (communication) cost of supporting multihoming / mobility under the architectures and protocols described in Section 2. For the LISP architecture, we also analyze extended variants that employ caching for EID-to-RLOC mappings, or Mobile-IP running over basic LISP.

3.1 Assumptions and Cost Definitions

We assume a single source-destination model where the source sends data packets at a constant rate. We analyze the average cost of managing a single interface (path) change to the destination, whether the interface change is due to re-routing to the multihomed destination or due to the mobility of the destination node.

The cost of delivery of a single packet is denoted by C_D. The total cost per interface change, denoted by C_{Tot}, is a function of the *location lookup* cost (C_L), *location update* cost (C_U), and *location inconsistency* cost (C_I). *Location lookup* cost is defined only for LISP, to capture the cost of querying a mapping server (Map Server) for information about the destination's RLOC given the destination's EID. *Location update* cost captures the cost of updating the location (routing) information of the destination node. In computing the *location inconsistency* cost, we assume that packets delivered to the wrong location due to inconsistency of location / routing information, need to be delivered again.

3.2 Model Parameters

In our model, we assume that the inter-arrival times of data packets and the lifetime of the destination's interface, each follows an exponential distribution, denoted by $f_p(t)$ and $f_m(t)$, respectively. We define the following two parameters:

- λ: the mean packet arrival rate, i.e., $f_p(t) = \lambda e^{-\lambda t}$.
- μ: the rate at which the interface to the destination changes, i.e., $f_m(t) = \mu e^{-\mu t}$.

Assuming that both packet arrival and interface lifetime processes are independent, the mean number of data packets received by the destination per a single interface change is given by: $\rho = \frac{\lambda}{\mu}$.

We define P to be the probability that the source has the correct (i.e., consistent) location / interface information. For example, under MIP, P defines the probability that the home router contains consistent routing / location information. Under LISP, P defines the probability that the cache or the Map Server contains correct routing information. Under RINA, P defines the probability that the DIF contains correct routing information.

Parameters/Costs	Definitions
λ	sending rate of the source
μ	mobility rate of destination or rate of interface failure for multihomed destination
ρ	$\frac{\lambda}{\mu}$
C_L	Cost of lookup
C_U	Cost of location update
C_D	Cost of delivery
C_I	Cost of inconsistency

Table 1: Definitions of Parameters and Costs

In steady state, P can be defined as the probability that the interface to the destination has not changed since the last packet delivery. Let t_p be the exponential random variable representing the packet inter-arrival time, and t_m be the exponential random variable representing the residual time during which the interface to the destination node does not change[2]. Thus, we have:

$$\begin{aligned} P &= Prob(t_p < t_m) \\ &= \int_{t_p=0}^{\infty} f_p(t_p) \int_{t_m=t_p}^{\infty} f_m(t_m) dt_m dt_p \\ &= \int_{t_p=0}^{\infty} \lambda e^{-\lambda t_p} \int_{t_m=t_p}^{\infty} \mu e^{-\mu t_m} dt_m dt_p \\ &= \frac{\lambda}{\lambda+\mu} \end{aligned}$$

The total cost per destination's interface change, C_{Tot}, is given by:

$$\begin{aligned} C_{Tot} \;=\; & C_L \;+\; C_U \;+ \\ & \rho(P \times C_D + (1-P) \times C_I) \end{aligned} \tag{1}$$

where C_I is defined as $(C_D + C_D^{OLD})$, and C_D^{OLD} is the cost of packet delivery to the old location / interface. Henceforth, we take $C_D^{OLD} = C_D$, assuming that packets delivered to the wrong location need to be re-delivered to the correct location at the same cost. We note that the first term, C_L, reflects the cost of EID-RLOC lookup in the LISP-Cache architecture. Under caching, this lookup cost is incurred only once per each destination's interface change, since subsequent packets will readily use the cached (correct) mapping at negligible cost. On the other hand, without caching, C_L is included in C_D, the delivery cost of every packet.

Table 1 summarizes our parameters.

3.3 MIP

For MIP, we define the cost terms in Equation (1) as follows:

- $C_L = 0$,
  since in MIP, the home router readily maintains the location of the destination node, and does not look up any mapping service.

---

[2] Recall that the residual time of an exponentially distributed time is also exponential due to the memoryless property.

- $C_D = C_{SN-HR} + C_{HR-DN}$,
  where the cost of delivery of a single packet, C_D, is the sum of C_{SN-HR}, which represents the cost of delivering a packet from the source node (SN) to the home router (HR), and C_{HR-DN}, which represents the cost of delivering the packet from HR to the destination node (DN).

- $C_U = C_{DN-FR} + C_{FR-HR}$,
  where the cost of updating the destination's interface / location is the sum of C_{DN-FR}, which represents the cost of updating the foreign router, and C_{FR-HR}, which represents the cost of updating the home router.

The costs of delivery under the MIP protocol are highlighted in Figure 1.

3.4 LISP

Figure 2 highlights a sequence of messages needed to deliver a data packet under LISP, assuming no caching of EID-RLOC mapping information. A source starts by sending the data packet through its Tunnel Router (TR). The source TR queries a Map Server (MS) for information about the destination TR. The source TR then forwards the data packet to the destination TR, which in turn forwards it to the destination node.

If the destination's interface changes due to mobility of the destination node from one autonomous system / domain to another, the destination node registers with the TR of that domain. Figure 4 highlights the sequence of messages needed for a destination node to update its location with the MS.

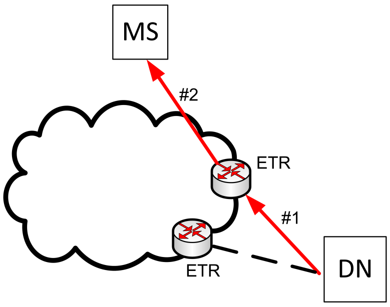

Figure 4: LISP cost of update

If the destination node were multihomed, the map server returns more than one EID-RLOC mapping that correspond to each of the destination's connections / interfaces. The source TR then chooses one of those mappings, and proceeds to send the data packet to the (chosen) destination TR. If this destination's interface fails, the source TR continues to use that failed interface until the corresponding mapping information in the map server is invalidated, and the source TR switches to a new (operational) destination's interface [7].

Thus under LISP, we define the cost terms in Equation (1) as follows:

- $C_D = C_L + C_{SN-DN}$,
  where the lookup cost, C_L, represents the cost of querying an EID-RLOC mapping server to identify the location of the destination TR. This lookup cost is incorporated in the delivery cost of every single data packet since we assume in this basic variant, that the location information is not cached at the source TR.

- $C_U = C_{DN-TR} + C_{TR-MS}$,
  where C_U, the cost of updating the Map Server (MS), is the sum of C_{DN-TR}, which represents the cost of location update from the destination node to its TR, and C_{TR-MS}, which represents the cost of updating the MS.

3.5 LISP-Cache

Iannone et al. [5] studied the use of caching at the source Tunnel Router under LISP. Naturally, caching would decrease the per-packet cost of looking up the EID-RLOC mapping information, as long as the cached location information is accurate. The packet delivery process is still the same as that of Figure 2 with the only difference being that the lookup is only done once per cache entry lifetime (which, we assume, corresponds to the expected inter-failure time of the destination's interface). Thus we define the cost terms in Equation (1) as follows:

- $C_L > 0$,
  which represents the cost of querying an EID-RLOC mapping server to identify the location of the destination TR. This lookup is done once whenever the destination's interface changes and then cached for subsequent data packets.

- $C_D = C_{SN-DN}$,
  where we assume that looking up the cache for the location information is negligible, and thus does not contribute to the cost of delivery of every single packet.

- $C_U = C_{DN-TR} + C_{TR-SN_{cache}}$,
  where C_{DN-TR} represents the cost of location update from the DN to its TR, and $C_{TR-SN_{cache}}$ represents the cost of invalidating the source TR's cache due to the change in the destination's interface.

3.6 LISP-MIP

Farinacci et al. [3] propose the use of MIP as a means to managing mobility in LISP. As mentioned earlier, mobility is a dynamic form of multihoming. Thus, this LISP-MIP variant can be generally used to deal with a change of destination's interface whether because of mobility or re-routing to a multihomed destination.

Figure 5 highlights the cost of message delivery under the LISP-MIP architecture. The source is sending a packet to the destination node that has already moved to another AS and got a new care-of-address and updated its home agent, following the MIP protocol. Once the home agent intercepts the message, it tunnels it to the new location. An additional lookup is needed to obtain the address of the current destination tunnel router.

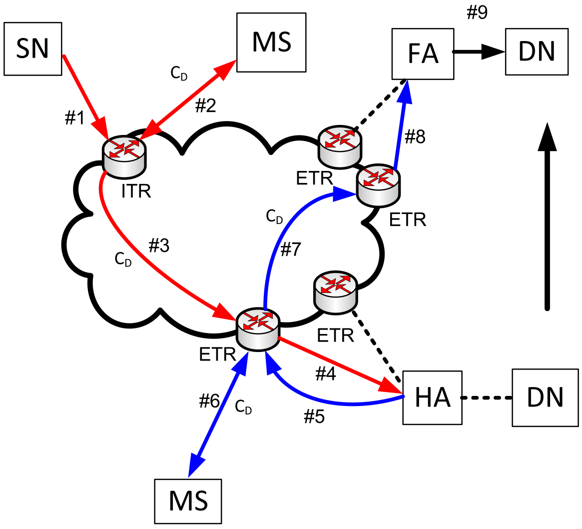

Figure 5: LISP-MIP cost of packet delivery

Thus under LISP-MIP, assuming no caching of location information, we define the cost terms in Equation (1) as follows:

- $C_D = C_{SN-L} + C_{SN-HR} + C_{HR-L} + C_{HR-DN}$,
  where the cost of delivery of a single packet includes C_{SN-L}, which represents the cost of querying a mapping service at the source's TR, and C_{HR-L}, which represents the cost of querying a mapping service at the destination's home TR.

- $C_U = C_{DN-FR} + C_{FR-HR}$,
  where the cost of updating the location is the sum of C_{DN-FR}, which represents the cost of updating the foreign router, and C_{FR-HR}, which represents the cost of updating the destination's home router.

3.7 LISP-MIP-Cache

As a last LISP variant, we augment the LISP-MIP model defined above with caching to reduce the cost of looking up location information. The delivery process still follows the same pattern as shown in Figure 5, the only difference is that the lookup is only done once per cache entry lifetime (which, we assume, corresponds to the expected inter-failure time of the destination's interface). We define the cost terms in Equation (1) as follows:

- $C_L = (C_{SN-L} + C_{HR-L}) > 0$,
  which represents the costs of querying a mapping server at the source's TR and the destination's home TR, respectively. We note that these lookup costs are only incurred once whenever the destination's interface changes. The location information is then cached for future use. Thus these lookup costs do not contribute to the delivery cost of every single data packet.

- $C_D = C_{SN-HR} + C_{HR-DN}$,
  which defines the cost of delivery of a single packet. The cost of looking up the cached location information is assumed to be negligible.

- $C_U = C_{DN-FR} + C_{FR-HR}$,
  which defines the cost of updating the destination's location at its home router.

3.8 RINA

Support for multihoming and mobility is inherent in the RINA architecture [1]. As reviewed earlier, a data packet is delivered hop-by-hop to the destination across limited-scope Distributed Inter-process communication Facilities (DIFs). If the destination's interface changes, then at the last routing hop, the mapping from the destination node's address to the new interface is locally propagated. This local update involves unsubscription / withdrawal from/of the old interface (underlying DIF), and subscription / registration to/of the new interface (underlying DIF), which in turn result in updating the routing information to map to the new interface.

Thus under RINA, we define the cost terms in Equation (1) as follows:

- $C_L = 0$,
  since in RINA, each node (process) readily maintains the next-hop (routing) information to the destination node, and does not look up any mapping service.

- $C_D = C_{SN-DN}$,
  since RINA strives to maintain a "direct" route to the destination.

- $C_U = C_{DIF-UNREG} + C_{DIF-REG}$,
  where $C_{DIF-UNREG}$ represents the cost of unsubscription from an underlying DIF that no longer leads to the destination, and $C_{DIF-REG}$ represents the cost of subscription to the new underlying DIF that leads to the destination.

A summary of the costs under all schemes is shown in Table 2.

Costs	Mobile IP	RINA	LISP	LISP-Cache	LISP-MIP	LISP-MIP-Cache
C_D	$C_{SN-HR} + C_{HR-DN}$	C_{SN-DN}	$C_L + C_{SN-DN}$	C_{SN-DN}	$C_{SN-L} + C_{SN-HR} + C_{HR-L} + C_{HR-DN}$	$C_{SN-HR} + C_{HR-DN}$
C_U	$C_{DN-FR} + C_{FR-HR}$	$C_{DIF-UNREG} + C_{DIF-REG}$	$C_{DN-TR} + C_{TR-MS}$	$C_{DN-TR} + C_{TR-SN_{CACHE}}$	$C_{DN-FR} + C_{FR-HR}$	$C_{DN-FR} + C_{FR-HR}$
C_I	$C_D + C_D^{OLD}$	$C_D + C_D^{OLD}$	$C_D + C_D^{OLD}$	$C_D + C_D^{OLD}$	$C_D + C_D^{OLD}$	$C_D + C_D^{OLD}$
C_L	0	0	$C_{TR-MS} + C_{MS-TR}$	$C_{TR-MS} + C_{MS-TR}$	$2(C_{TR-MS} + C_{MS-TR})$	$2(C_{TR-MS} + C_{MS-TR})$

Table 2: Costs

4. NUMERICAL RESULTS

We present numerical results based on the cost equations defined in Section 3. As mentioned earlier, we define costs in terms of average path lengths between communicating entities, e.g., between a source's TR and a mapping server in LISP. Thus, we present results for two kinds of network topologies: grid and Internet-like.

4.1 Grid Topology

For an $N \times N$ grid topology, the average distance between any two nodes is given by $1.333(N/2)$ hops. We use this average distance as the cost of communication between two nodes that are not on the same network. On the other hand, if the communicating nodes are on the same network, the cost is relatively smaller (and independent of the size of the topology) — we take the cost to be two hops between a node and its TR, and one hop otherwise.

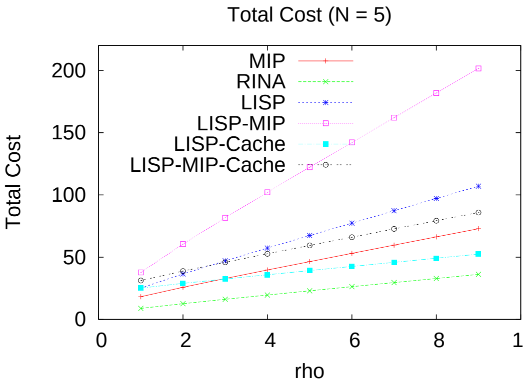

Figure 6: Numerical results for a 5×5 grid

Figure 6 shows the total costs of the various schemes as ρ takes values greater than one. As ρ increases, the total cost for all schemes increases (as expected). We observe that as ρ increases, plots for schemes with and without caching intersect as caching starts to yield less cost since cached location information becomes more accurate for a larger (average) number of packets delivered before the destination's interface changes.

RINA has the lowest total cost, while LISP-MIP has the worst cost. LISP-MIP incurs LISP's overhead of lookup cost at the source and at the home of the destination node, in addition to MIP's overhead of tunneling the packet from the home node to the destination node.

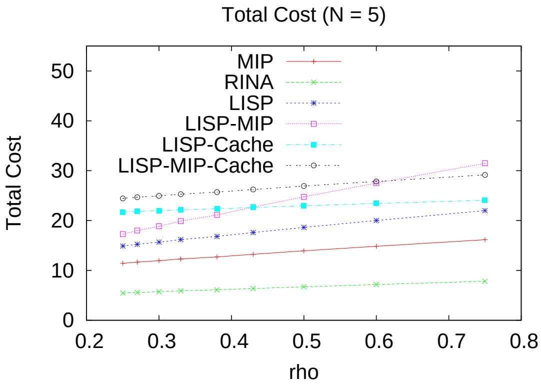

Figure 7: Numerical results for a 5×5 grid

Figure 7 shows the total costs of the various schemes as ρ takes values less than one. Again, as ρ increases, the total cost for all schemes increases, but LISP-MIP's cost increases relatively faster due to the aforementioned overheads of both LISP and MIP.

Figure 8 shows the total costs of the various schemes for varying grid sizes N for $\rho = 2$. As N increases, the total cost for all schemes increases, with RINA incurring the lowest cost at a sublinear increase rate.

4.2 Internet Topology

To consider Internet-like topologies, we take the average path length to be 15 hops when computing costs, to match findings from Internet measurement studies [9]. Figure 9 shows the results, which are consistent with those of Figure 6. We observe, however, that RINA's

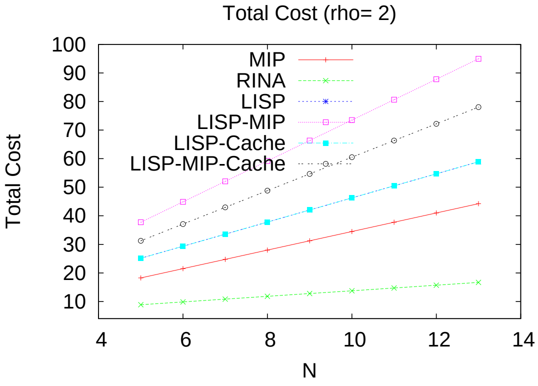

Figure 8: Numerical results for varying grid sizes

cost is even relatively lower because it maintains a "direct" route to the destination by locally adapting to interface changes, resulting in a more pronounced advantage when average path length is larger.

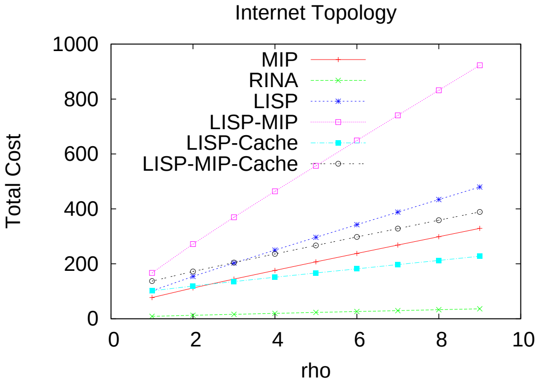

Figure 9: Numerical results for Internet topology

5. TRACE-DRIVEN SIMULATION

In this section we validate our analytical results using trace-driven simulation based on real packet traces from CAIDA [12].

DataSet	Chicago	San Jose
Unique ASes	66	97
Packets	74123	74123
Nodes	2178	2425
Edges	4488	5041

Table 3: Topology Properties

5.1 Experimental Setup

We base our simulations on CAIDA's anonymized packet traces [12]. Our simulation considers only multihoming, so we do not include experimental results for Mobile-IP. We select two datasets from two Equinix locations: Chicago and San Jose (dated 20090219-045912 and 20090219-060100, respectively). The traces consist of anonymized tcpdump packets from different source-destination pairs. Each trace file contains more than a million records. Since the traces provide only source-destination pairs and packet arrival times, we use the BRITE topology generator [6] to generate an underlying AS and router network topology.

We use the top-down generation model of BRITE which is based on two phases. In the first phase, unless otherwise specified, an AS topology is initially generated using the Waxman model[3]. In the second phase, a router-level topology is generated for each AS. Router nodes are placed randomly on the 2D-plane and connected using the Waxman model. The average path length between nodes in the generated topologies is 14 hops, consistent with Internet measurement studies [9]. To keep the simulation and generated topologies manageable, we only consider the first 74123 packets from each packet trace, and make the simplifying assumption that all IP addresses which have a common 16-bit prefix belong to the same AS. Table 3 highlights properties of our two simulated topologies.

We utilize the packet timestamp as the packet arrival time. For LISP, we assume that updating the map server takes an exponentially distributed time with a mean value that corresponds to the average path length—for simplicity, we assume the delay of one hop is 1 ms. For RINA, we assume the time it takes the "ancestor" node to update its next-hop is exponentially distributed with mean of 2 × RTT, where RTT is the round-trip time between the ancestor node and destination node. We take the ancestor node to be the node closest to the destination that is common to the shortest path and second shortest path from the source.

Furthermore, the time between link failures follows an exponential distribution. To simplify our simulation model, we assume that a single link (interface) fails at a time. We also make sure that interface failures occur only on destinations that are multihomed.

5.2 Results

In this section, we present the results of our simulations. We measure packet drop ratio, and packet delivery delay. All results are presented with 90 percent confidence intervals. Figures 10 and 11 show the re-

---

[3] AS nodes are assumed to be distributed on the 2D-plane according to a heavy-tailed distribution, just for the purpose of using distances in the Waxman model to generate connectivity.

sults of packet drop ratio using simulations based on the two datasets. The results confirm our analytical model. RINA drops around 2% and 2.5% of the packets, respectively, while BGP, LISP, and LISP with caching, drop around 4% and 8% of the packets, respectively.

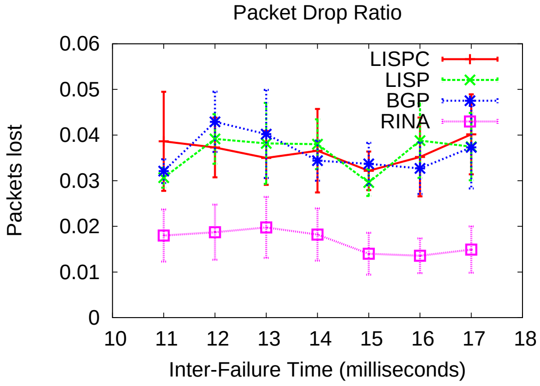

Figure 10: Packet Drop Ratio (Chicago dataset, Waxman AS-topology)

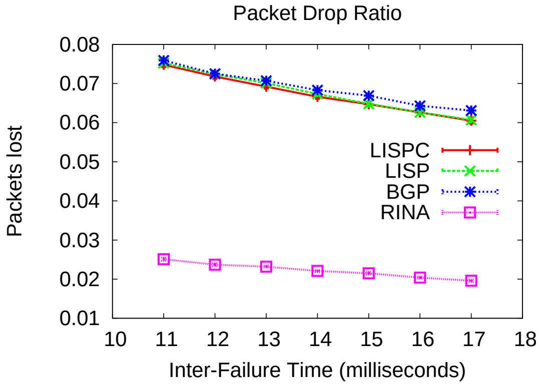

Figure 11: Packet Drop Ratio (San Jose dataset, Waxman AS-topology)

Figures 12 and 13 show the average packet delivery time. The delivery time of RINA and BGP is smaller due to the fact that there is no need to contact a map server. The benefit of caching for LISP is highlighted by smaller average delivery time.

Note that BGP's delay is slightly lower than that of RINA, since BGP's lack of fine-grained routing control makes it incapable of adapting to a failure of the shortest path to a specific destination node, however, for those packets that get delivered when the shortest path is up, their delivery delay is smallest. On the other hand, RINA enables the construction of an "overlay" network between the source node, destination node, and an intermediate node (ancestor) that is capable of re-routing around failed paths (interfaces). Thus, under RINA, more packets are successfully delivered, but those packets taking alternate paths when the primary paths are down, experience slightly higher delay.

We also observe that under LISP, the delay is almost double that of RINA and BGP, since LISP requires a mapping lookup which adds extra delay that is in the order of the average path length of around 14 hops (msec) in our topologies.

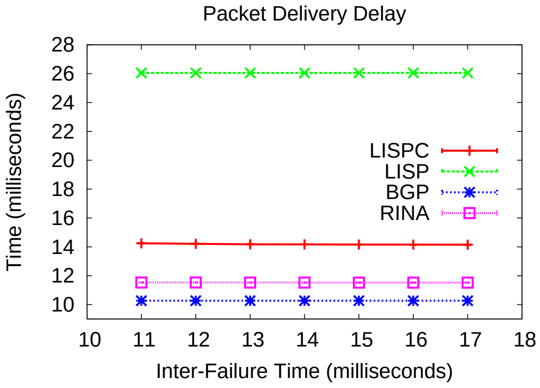

Figure 12: Average Packet Delivery Time (Chicago dataset, Waxman AS-topology)

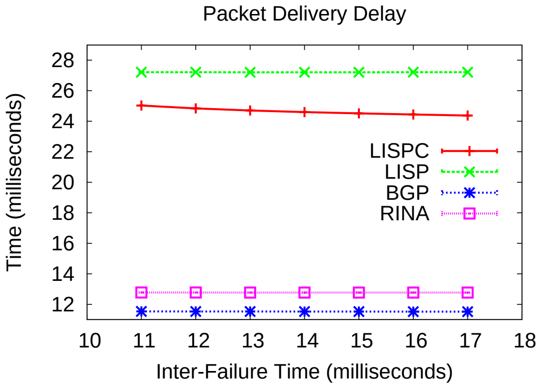

Figure 13: Average Packet Delivery Time (San Jose dataset, Waxman AS-topology)

We also experimented using topologies generated using BRITE's top-down approach, where in the initial phase, the AS topology is generated using the Barabasi-Albert (BA) model with incremental growth type and preferential connectivity. Results are shown in Figures 14 and 15 for the San Jose dataset, and in Figures 16 and 17 for the Chicago dataset. The results are consistent with our Waxman AS-topology results. RINA yields the lowest cost in terms of packet drop ratio, delivering packets at the lowest possible delay due to its local routing adaptation within the scope of the overlay involving the source, destination, and "ancestor" node.

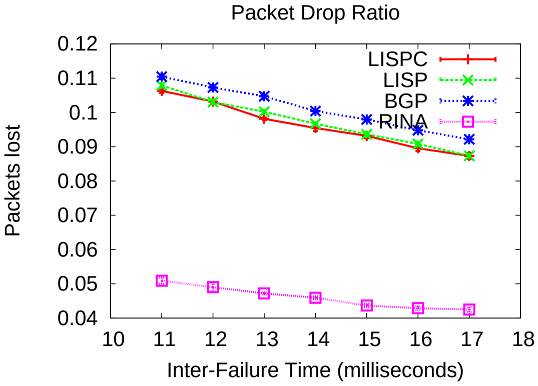

Figure 14: Packet Drop Ratio (San Jose dataset, BA AS-topology)

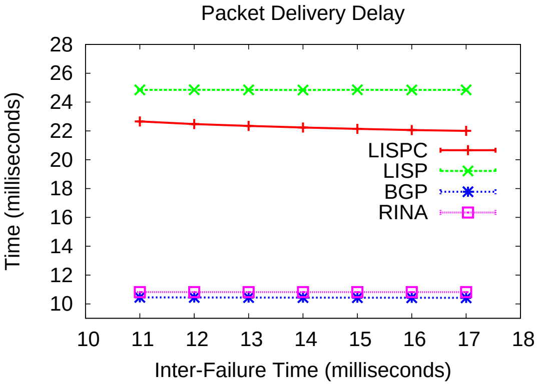

Figure 15: Average Packet Delivery Time (San Jose dataset, BA AS-topology)

6. CONCLUSION

We developed a cost model to evaluate the multihoming / mobility support of RINA, LISP, MIP, and variants with and without caching of location / routing information. RINA incurs the lowest cost, while LISP-MIP incurs the highest cost. We also validated our model using simulation on real packet traces.

7. ACKNOWLEDGMENT

We would like to thank John Day for various discussions on aspects of this work and for his encouragement.

This work has been partially supported by National Science Foundation awards: CISE / CCF #0820138, CISE / CSR #0720604, CISE / CNS #0524477, CNS / ITR #0205294, and CISE / EIA RI #0202067.

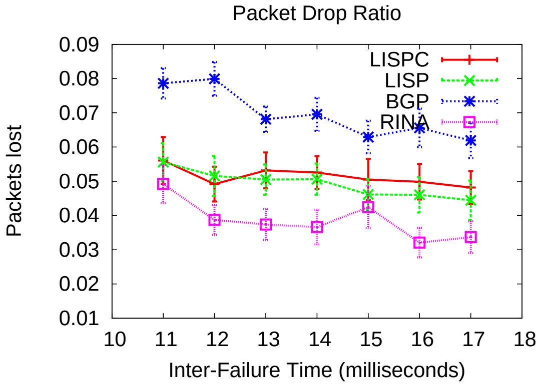

Figure 16: Packet Drop Ratio (Chicago dataset, BA AS-topology)

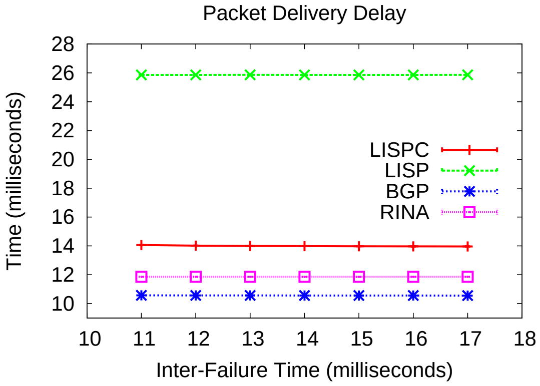

Figure 17: Average Packet Delivery Time (Chicago dataset, BA AS-topology)

8. REFERENCES

[1] John Day, Ibrahim Matta, and Karim Mattar. "Networking is IPC": A Guiding Principle to a Better Internet. In Proceedings of ReArch'08, Madrid, SPAIN, December 2008.

[2] D. Farinacci and V. Fuller. LISP Map Server, March 2009. draft-fuller-lisp-ms-00.txt.

[3] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis. Locator/ID Separation Protocol (LISP), March 2009. draft-farinacci-lisp-12.txt.

[4] L Iannone, D Saucez, and Olivier Bonaventure. OpenLISP: An Open Source Implementation of the Locator/ID Separation Protocol. In IEEE INFOCOM, Demo paper, April 2009.

[5] Luigi Iannone and Olivier Bonaventure. On the Cost of Caching Locator/ID Mappings. In Proceedings of the ACM CoNEXT '07 conference, pages 1–12, New York, NY, USA, 2007. ACM.

[6] Alberto Medina, Ibrahim Matta, and John Byers. BRITE: A Flexible Generator of Internet Topologies. Technical report, Boston, MA, USA,

2000. http://www.cs.bu.edu/brite.

[7] D. Meyer and D. Lewis. Architectural Implications of Locator/ID Separation, January 2009. draft-meyer-loc-id-implications-01.txt.

[8] David Meyer. The Locator/Identifier Separation Protocol (LISP). http://www.1-4-5.net./~dmm/lisp/.

[9] Vern Paxson. End-to-end Routing Behavior in the Internet. SIGCOMM Comput. Commun. Rev., 36(5):41–56, 2006.

[10] Charles Perkins. IP Mobility Support for IPv4. RFC 3344, Internet Engineering Task Force, August 2002.

[11] Bruno Quoitin, Luigi Iannone, Cédric de Launois, and Olivier Bonaventure. Evaluating the Benefits of the Locator/Identifier Separation. In MobiArch '07: Proceedings of 2^{nd} ACM/IEEE International Workshop on Mobility in the Evolving Internet Architecture, pages 1–6, New York, NY, USA, 2007. ACM.

[12] Colby Walsworth, Emile Aben, kc Claffy, and Dan Andersen. The CAIDA Anonymized 2009 Internet Traces. www.caida.org/data/passive/passive_2009_dataset.xml.