(Institute for Systems Research Technical Report: TR-2000-5)

An Approach to Fixed/Mobile Converged Routing

M. Scott Corson
corson@isr.umd.edu
Institute for Systems Research
University of Maryland
College Park, MD 20742

Alan O'Neill
alan.w.oneill@bt.com
BT Laboratories, Adastral Park
British Telecommunications, PLC
Ipswich, Suffolk, IP5 3RE

Abstract

We consider a family of routing protocols for networks in which the core topology is essentially fixed but where the end systems may be mobile. We refer to this form of routing as Fixed/Mobile Converged (FMC) routing. This is a mixture of the traditional prefix-routed scenario of the fixed Internet, and the classical edge mobility scenario that is today supported by cellular networks, primarily as part of the cellular technology elements (GSM, GPRS, etc). We outline a general architecture for the support of such edge mobility, and present an approach to FMC routing that fits within this architecture. We then present initial simulation results illustrating the potential scalability and routing efficiency of this approach.

1. Introduction

Telecommunications networks are rapidly transitioning towards an all-IP architecture. This trend is not restricted to fixed networks. Second generation cellular networks have been modified to provide limited data services, and third generation systems undergoing rollout now have been designed to deliver IP connectivity to the end user.

Projecting technological trends, it is seen as inevitable that the future global telecommunications network will consist of a routed/switched IP core (much of which will be optical) accessed via a wide range of edge technologies. Many of these edge technologies will support mobility based on the continuing advancements in wireless technology. Internet service providers will increasingly want to support both fixed and mobile users. With IP routing technology being pushed out to the network's edge, it becomes less cost effective to support the various modes of layer 2 edge mobility management that accompany today's cellular and PCS technologies. Rather, unified solutions become advantageous to domain operators, wherein mobility management becomes an integral component of an IP layer routing protocol.

Internet routing protocols have been traditionally designed from an assumption that the location of an IP interface in the topology is static. In addition, they assume that address allocation within the topology will aim to provide multiple levels of IP address aggregation such that routing protocols can deal with address prefixes, rather than large numbers of host routes. Within this framework, traditional intra-domain protocols, such as OSPF, need only react to infrequent changes to the network due to link or router failures or permanent modifications to the addressing scheme or the topology.

Mobile Ad hoc NETwork (MANET) routing protocols have been developed to address what could be considered to be an extreme scenario, whereby the mobile nodes have permanent IP addresses and can rapidly roam through an ad hoc topology, leading to the need for alternative routing technology and the general loss of aggregation opportunities.

This paper considers a third family of routing protocols for the case in which the core topology is essentially fixed but where the end systems *may* be mobile. This is a mixture of the traditional prefix-routed scenario of the fixed Internet and the classical edge mobility scenario that is today supported by cellular networks, primarily as part of the cellular technology elements (GSM, GPRS etc). Migrating the latter mobile routing functionality to layer 3—to release all the end-to-end internetworking benefits which has aided the deployment of the Internet—would tend to suggest a *fusion* of the MANET and traditional routing protocol architectures. The primary aim is to move the IP interface location in the routing topology as the mobile changes base stations so that active IP sessions are maintained. We refer to this form of routing as Fixed/Mobile Converged (FMC) routing within an Edge Mobility Architecture [EMAdraft].

These networks can be considered to have a single IP routing protocol that runs between routers in the FMC domain. Some of these routers may be Network Access Points (NAPs) for collections of fixed hosts (e.g. PSTN or cable head-ends). Others may be Base Stations (BSs) equipped with a (potential) diversity of wireless technologies such as CDMA, TDMA and Radio LANs etc. The radio layers are assumed to provide the well known layer 2 handover models and other capabilities including break-before-make, make-before-break, power measurement, mobile-assisted handover and security features. To facilitate internetworking, inter-base station coordination is assumed to use IP-based communication using messages which are abstractions of the messages which are today carried in cellular technology-specific messages, often via central processing elements.

The remainder of this paper is structured as follows. In section 2, we describe an Edge Mobility Architecture (EMA) for the generic support of edge mobility, with the aim of being general enough to support a range of different routing protocols, as well as enabling handover between diverse types of cellular technology through capability exchange between radio-equipped BSs. This paper does not discuss the effects of the architecture on DNS, DHCP and infrastructure services, nor does it contribute to the debate on the appropriate layer and model for mobile terminal paging. It will become clear that the routing approach put forth here needs to be coupled with a companion paging architecture for location management, but this will be the topic of a future paper. In section 3 we provide a detailed description of our proposed approach for Fixed/Mobile Converged (FMC) routing, and compare it with alternative proposals for IP mobility support in section 4. We then present an initial performance analysis in section 5. Finally, we give some concluding remarks in section 6.

2. Edge Mobility Architecture

The architecture proposed assumes that modifications to either MANET or traditional routing protocols are possible which will enable these protocols to comply with this architecture and hence facilitate a message set and control model which has a degree of protocol independence.

The architecture has seven main components, the first being the use of Mobile IP across provider boundaries to facilitate the temporary movement of an IP address (on a mobile terminal interface) away from its home domain whilst maintaining active sessions.

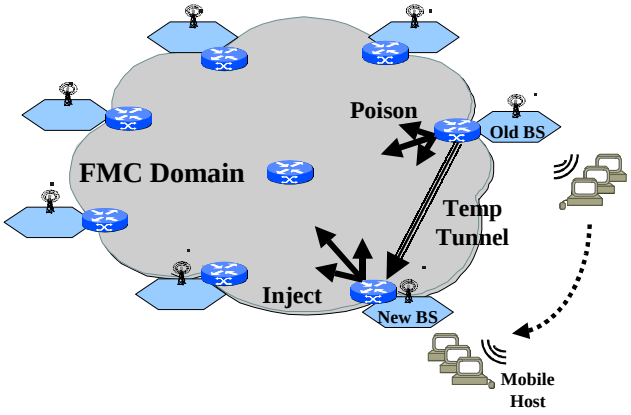


Figure 1: EMA Handover within an FMC Domain

The mobile IP tunnels are initiated by a BS Home Agent (HA) at the inter-domain boundary to the BS Foreign Agent (FA) in the foreign domain, and are extended to further BSs in the foreign domain using normal Mobile IP FA mechanisms. This bounds subsequent discussions to intra-domain routing issues. The other six components are:

1) the provision of a modified intra-domain routing protocol which provides prefix-based routing within a domain, with each **prefix** representing a **block** of IP addresses allocated to each NAP or BS in the domain, as well as **host routes** to support **mobile host migration** away from the allocating (IP address) BS;
2) the provision and use of virtual links for routing exchange and messaging between cooperating BSs to exchange capabilities, and to effectively and locally manage the handover of the responsibility for, and routing of, the mobile terminal and it's associated IP address;
3) the provision and use of a **temporary tunnel** to redirect packets in flight between the old BS and the new BS whilst routing converges;
4) the ability to **inject** a host route for the mobile;
5) the ability to **poison** the existing route to the mobile; and
6) a method to return the allocated IP address to the allocating BS on mobile session termination at a different BS in the same domain.

The reasons for each of these components will be explained in the following subsections that give examples for CDMA (i.e. make-before-break) and TDMA (i.e. break-before-make) handover.

2.1 Mobile Session Start-Up

The mobile connects to the nearest/best BS and is brought into the IP routing domain by requesting and being allocated an IP address out of the block of addresses managed by that BS. This **allocating BS** (ABS) will be advertising the IP address prefix associated with that address block into the intra-domain routing protocol such that "at home" mobiles have a proactively and permanently advertised route, and are immediately reachable to all hosts in the internet. Note that end hosts statically attached to the FMC domain via NAPs can be viewed as "at home" mobiles who never move. When a mobile changes BSs, its IP address(es) move(s) with it so that higher-layer sessions are unaffected. This is accomplished by modifying the intra-domain routing using hosts routes to overrule (longest match) or overwrite the underlying, proactive prefix routing to the allocating BS. Placing an appropriate set of messages over IP ensures that a wide range of radio technology-specific handover models can be accommodated within a single IP model to allow for internetworking of IP over those diverse technologies.

2.2 Break Before Make

TDMA-based cellular and micro-cellular systems require a **break** in communications with the **old BS** (OBS) whilst a **mobile host** (MH) undergoes handover before a subsequent **make** operation establishes layer 2 connectivity at a **new BS** (NBS). The following steps outline the break-before-make (BBM) procedure of the EMA:

1) Detect imminent handover and inform new and old BSs (optional).
2) Build a temporary tunnel from old BS to new BS (optional).

(Break event occurs)

3) Await Make event at new BS.
4) On Make event, inject new route to the new BS and poison old route to old BS.
5) Tear down tunnel at old BS (if present).

The first two steps are optional because it is not always possible to predict a handover in advance. An unanticipated link failure may occur between the old BS and the mobile host prior to its arrival at the new BS. The "inject/poison" route features of the EMA can be invoked only *after* the old and new BSs have been identified, and have agreed to the handover by creating the dynamic, temporary redirect tunnel between them. Note that for efficiency purposes a single redirect tunnel could be pre-configured between adjacent base stations to support all inter-base station handovers, and dynamic mobile-specific redirect tunnel state temporarily installed against that aggregate tunnel.

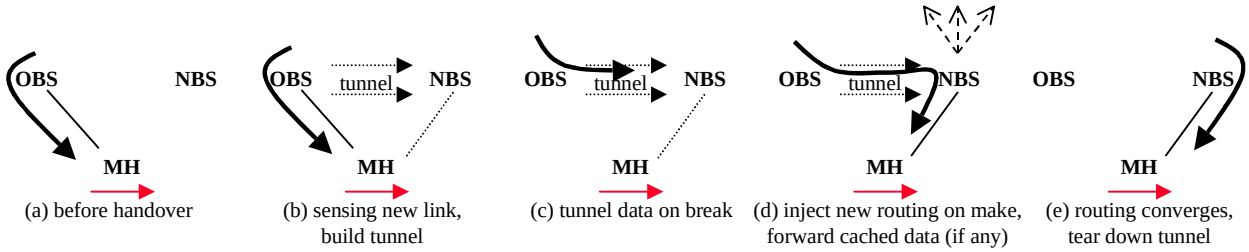


Figure 2: Basic EMA BBM Handover (with temporary tunnel)

TDMA technology such as GSM only allows the mobile to be connected to a single BS at a time, with a data path dead-time incurred during handover. To minimize the potential for packet loss while maintaining efficient routing, the inject/poison route features are delayed and invoked only *after* the Make event occurs at the NBS thereby ensuring that the link to the OBS is used until the break occurs. When the mobile disconnects at the radio layer from the old BS (Break), the new BS, through the inter-BS virtual link or tunnel (if present), is immediately known to be the next best hop, and packets hitting the old BS are immediately redirected down the tunnel to the new BS. If a tunnel is not present (unanticipated break), then packets may be dropped. Some time later the mobile will attach to the new BS (Make) and, if the tunnel is in place, will immediately receive in-flight and locally cached packets.

Once the Make event occurs, the new BS informs the new BS of the need to commence handover. The two BSs now collaborate to locally inject the new route into the routing domain and poison the old route. Packets to the mobile will then head towards the new BS route. This route redirect process will typically converge during the data path dead-time ensuring that only a small number of packets (if any) will need to be tunneled from old to new BS. Once routing has converged, the old BS will eliminate the redirect state associated with the temporary tunnel. The reception of the mobile at the new BS is then confirmed through acknowledgement messages to the old BS which is used to confirm handover of responsibility for the mobile and it's IP address in the system.

2.3 Make Before Break

The following steps outline the make-before-break (MBB) procedure of the EMA:

1) Detect imminent handover and inform new and old BSs (optional).
2) Build a temporary tunnel from old BS to new BS (optional).
3) Await Make event.

(Make event occurs)

4) On Make event, inject new route to the new BS and poison old route to old BS.

(Break event occurs)

5) Tear down tunnel at old BS (if present) and remove state of poisoned route.

CDMA technology enables a mobile terminal to be connected to two BSs at the same time and to undertake measurements to establish the preferred channel and handover time. This handover time determines the timing of the Make event. As with TDMA, it is desirable to *wait* until the Make event occurs before injecting the new host route.

However, unlike TDMA, packets continue to arrive at the mobile via the link to the old BS prior to the Make event. Thus the temporary tunnel is typically not needed in CDMA, but it is constructed in case an unexpected early break occurs with the added benefit that in so doing the same handover state machine can be used for both BBM and MBB modes.

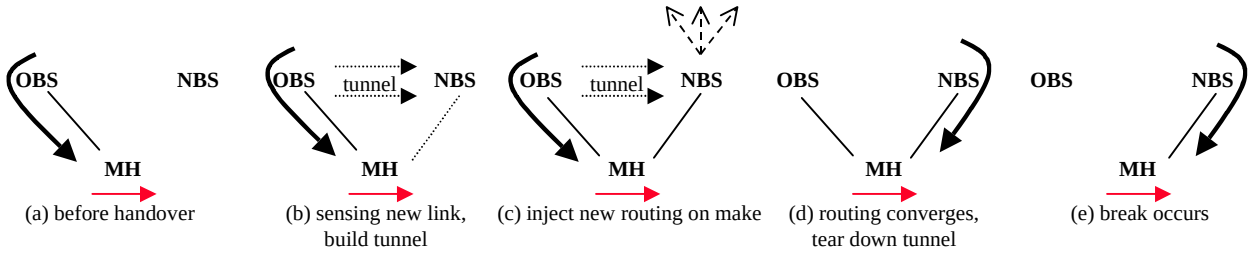


Figure 3: Basic EMA MBB Handover (with temporary tunnel)

We are not directly addressing IP layer support for CDMA soft-handover. While feasible in terms of IP layer routing, this mode of CDMA-based handover requires highly synchronized packet delivery over the air interface to the mobile which may not be compatible with a heterogeneous IP network infrastructure. Soft handover can still be achieved however if the underlying layer 2 (ATM AAL) has the required timing and cell duplication functionality as presently proposed in 3G systems. This is achieved for example by the IP layer at the OBS handing responsibility for soft handover to the ATM layer which duplicates and times cell delivery to the multiple neighbor NBS(s).

2.4 Hybrid Model

When a hybrid node is able to support both TDMA and CDMA (or other combinations of technology), then a consistent set of base station messages makes handover of the concurrent sessions between TDMA and CDMA base stations possible. This is achieved by the base stations understanding each other's capabilities, and holding up and synchronizing inject and poison stages as appropriate.

2.5 Mobile Switch Off

It is clear that the migration of IP addresses away from the allocating BS can lead to address exhaustion and a gradual degradation over time of the usefulness of the proactively-advertised BS address block prefix. It is therefore critical that at the moment that the mobile finishes active sessions, at a distant BS, that the IP address is returned to the home BS. This can be modelled as a virtual mobile moving from the distant BS back to the home BS and then locally returning the IP address. This can be accomplished using similar mechanisms which are used to support real inter-BS handovers, with the BSs acting as proxies for the virtual mobile. Their aim is to co-ordinate the removal of all host-specific routing entries in the domain as a result of previous mobility away from the home BS.

3. An Approach to FMC Routing

The preceding architecture does not specify how an FMC routing algorithm creates or modifies its host and prefix-specific IP forwarding entries, and various approaches are possible. With the objective of building a large-scale FMC domain, the Temporally-Ordered Routing Algorithm (TORA) appears potentially well-suited for use as a FMC routing algorithm [TORA]. TORA was originally conceived as a MANET routing algorithm where it is intended for use in large-scale, dynamic, bandwidth-constrained MANETs. The principle objective behind its design is the achievement of "flat scalability", i.e. achieving a high degree of scalability (measured as the number of routers in a domain) with a "flat", non-hierarchical routing algorithm. In its operation the algorithm attempts to suppress, to the greatest extent possible, the generation of far-reaching control message propagation.

With TORA, such suppression may or may not be feasible depending on the topology. As we will see, a key to achieving highly-scaled FMC routing with TORA turns out to be an issue of **topology control**. Under appropriate topological conditions, TORA's reaction to link additions and failures can be highly localized. This is a key property which we exploit based on the realization that, viewed abstractly, the "make" and "break" operations in cellular networks correspond to link "additions" and "failures", respectively, in a unified mobile host/fixed router network. The subsequent lack of large amounts of far-reaching control message propagation—a feature common to shortest-path algorithms—afford TORA its relatively quick convergence and consequent stability.

These properties appear desirable for the design of large-scale routed domains without any consideration for mobility support. In the most recent Internet Architecture Board (IAB) report on network layer issues [IAB99], the

IAB concluded that the scalability bottleneck of presently deployed routing technology stems not from storage considerations but rather from long convergence times. These convergence delays are due to the time required to distribute stable routing information updates (communication complexity) and the time required to re-compute routing tables (computational complexity). Operating in a suitable topology, TORA can have relatively low communication and computational complexity due to the nature of its distributed computation that forgoes shortest path computation.

TORA also supports loop-free, multipath routing realized as a consequence of the usage of temporally, totally-ordered "heights". The provision of multipath routing makes the protocol amenable for load sharing and traffic engineering. The algorithm also has the potential to support fast restore via its link reversal mechanism[1] based on the availability of fine-grained link status sensing (possibly from layer 2).

Recalling the EMA, the problem of FMC routing is divided into two sub-problems: inter-BS/NAP routing and host-specific routing.

3.1 Inter-BS/NAP Routing

Inter-BS routing is *prefix-based*, i.e. each BS/NAP advertises a prefix address covering a block of host addresses it controls. This routing should be continuously maintained, i.e. *more proactively*, whereas host-specific routing (to be discussed subsequently) should be maintained only as needed, i.e. *on-demand* or *reactively*.

3.1.1 TORA Concepts

TORA was originally specified as an on-demand routing algorithm, but this mode of operation is not generally required and a more proactive mode is possible. Because TORA proceeds independently for each destination, it may operate more proactively for certain destinations and reactively for others. In the proposed FMC context, separate versions of TORA will proceed more proactively for each BS/NAP and proceed reactively for each mobile host in an edge mobility-enhanced mode as necessary. Much of TORA's original protocol mechanism deals with reaction to link and node failures. Many of these details are not relevant to the discussion here, and we refer the reader to [TORA, TORAdraft] for this information. Here we focus on the aspects of TORA necessary to understand its operation as a FMC algorithm within the EMA.

Notation and Assumptions

We model our FMC network as a general graph $G = (N, L)$, where N is a finite set of nodes and L is a set of initially undirected links. Each node $i \in N$ is assumed to have a unique **Node ID** (NID), and each link $(i, j) \in L$ is assumed to allow two-way communication (i.e., nodes connected by a link can communicate with each other in either direction). Due to the mobility of some nodes, the set of links L is changing with time (i.e., new links can be established and existing links can be severed). From the perspective of neighboring nodes, a node failure is equivalent to severing all links incident to that node. Each initially undirected link $(i, j) \in L$ may subsequently be assigned one of three states; (1) undirected, (2) directed from node i to node j, or (3) directed from node j to node i. If a link $(i, j) \in L$ is directed from node i to node j, node i is said to be "upstream" from node j while node j is said to be "downstream" from node i. For each node i, the one-hop "neighbors" of i, $N_i \subset N$, is defined to be the set of nodes j such that $(i, j) \in L$. For the subsequent discussion, we assume each node i is always aware of its neighbors in the set N_i. This information is obtainable in practice from layer 2 mechanisms or a HELLO protocol. It is also assumed that all transmitted packets are received correctly within a finite time and in order of transmission. This is assumed initially as an aid to presentation, and deviations from this assumption will be noted as necessary. In practice, only some of the messages to be described require reliable delivery, and the requirements for reliability vary as well. A reliable messaging sub-layer protocol will be required for those messages requiring reliable, in-order, hop-by-hop delivery.

TORA builds and maintains a Directed Acyclic Graph (DAG) rooted at a destination. A destination identifier in TORA is a network **prefix**, composed of an interface IP address and a network mask. In a manner similar to OSPF, TORA uses a **Router ID** (RID) to uniquely identify a TORA router separately from its interfaces. Router IDs are unique identifiers internal to the routing algorithm, and are typically derived from the IP addresses of a router's interfaces. A TORA router exchanges HELLO packets with its one-hop neighbors via scoped multicasts to the AllTORARouters multicast address. A TORA router is typically multi-homed; its HELLO packet contains its RID and its interfaces' IP addresses and associated net masks. A TORA router may advertise multiple prefixes, and multiple DAGs may terminate at a TORA router—each maintained by a separate version of the algorithm. Unlike OSPF, knowledge of a router's RID is not disseminated beyond its one-hop neighbors. Only destination prefixes may be

[1] Due to space constraints we cannot give a complete description of the protocol's link reversal operation in response to arbitrary link failures (see [TORA,TORAdraft] for these details). We will only touch on aspects of the link failure processing as necessary.

disseminated over multiple hops, and these are not associated with a Router ID. As mentioned previously, TORA operates with respect to "nodes". A TORA **node** may be either a **router** or a **host**. A **Node ID** (NID) is a *polymorphic* identifier, and may be either a RID or a destination network prefix depending on the context. Consequently, the neighbor set N_i that lists a node's neighbors by NID may actually contain two different identifiers. A **neighbor** may be identified as a **router** (by its RID) or as a **destination** (by its network prefix) or frequently as both with multiple entries in the neighbor set table.

For a given destination, each participating node i is assigned a **height** defined as an ordered quintuple $H_i = (\tau_i, oid_i, r_i, \delta_i, i)$. No two nodes may have the same height (i.e. the set of heights is totally-ordered). Information may flow from nodes with higher heights to nodes with lower heights. Conceptually, information can be thought of as a fluid that may only flow downhill (see Figure 4). By maintaining a set of totally-ordered heights at all times, it is easy to see how loop-free, multipath routing is assured[2]. Information would have to flow uphill to form a loop, and this is not permitted. Height comparisons are performed *lexicographically*. Starting with the τ_i value, comparison tests are for "less than" or "greater than", with equality resulting in the comparison proceeding element-wise towards the final element, the NID i.

Conceptually, the height quintuple associated with each node is composed of two parameters: a **reference level**, and a **delta** with respect to the reference level. The reference level is represented by the first three values in the quintuple (a triple), while the delta is represented by the last two values (a double). A *new* reference level is defined each time a mobile host undergoes a *handover* with respect to the fixed network, or when a node loses its *last* downstream link due to a *link failure*. A reference level is "set to zero" when a DAG is *initialized*, or "reset to zero" when it is *optimized*.

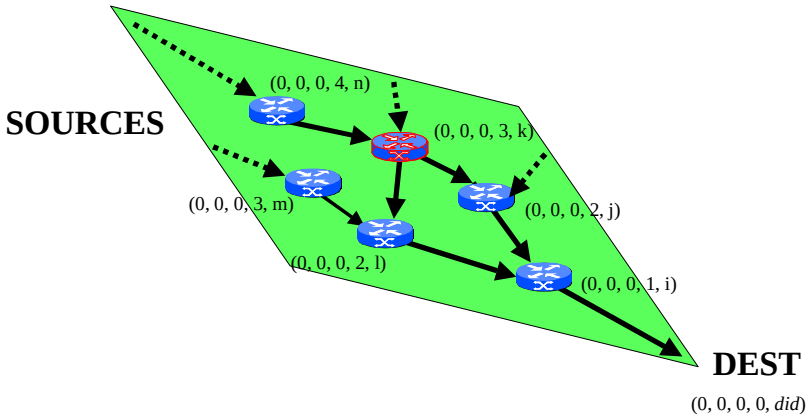


Figure 4: Downhill flow of information along DAG

The first value in the reference level, τ_i, has three meanings. If equal to zero, it indicates that the height value has remained "unchanged" since the DAG was initially constructed or was last optimized (this is the state of all heights in Figure 4). If positive, it is a time tag representing the "time" of a link failure somewhere in the network. If negative, it represents a route "freshness" value (the more negative the fresher the route) generated in response to handover-induced mobility. A reference level is referred to as **positive**, **zero** or **negative** depending on the sign of its τ_i value. The "originating RID" oid_i and "reflection bit" r_i values have significance (i.e. are non-zero) only for *positive* reference levels, and are not relevant to our discussion here.

The first value of the delta, δ_i, is an integer used to order nodes with respect to a common reference level. This value is instrumental in the propagation of a reference level. For heights with zero reference levels, its value approximately indicates the node's distance from the destination as shown in Figure 4. Its meaning for heights with positive reference levels is relevant only to the link reversal mechanism and is also beyond the scope of this paper. The second value of the delta, i, is the unique NID of the node itself. This ensures that nodes with a common reference level and equal values of δ_i (and in fact all nodes) can be totally ordered lexicographically at all times.

Each node i (other than the destination) maintains its height, H_i. Initially the height of each node in the network (other than the destination) is VOID, meaning that nodes do not have height or state information of any kind regarding the destination. They are unaware of the destination's existence. Subsequently, the height of each node i can be modified in accordance with the rules of the protocol. The height of the destination is always ZERO, $H_{did} = (0, 0, 0, 0, did)$, where did is the destination-ID (i.e., the unique Node ID equal to the destination network prefix for which the algorithm is running).

---

[2] As with many non source-routed protocols such as OSPF, transient loops may exist in the tables between adjacent routers that are exchanging routing control messages. With TORA these loops are only of the "ping pong" type, i.e. between adjacent routers, and are very short-lived in wired networks due to the speed and reliability of neighbor information exchange.

In addition to its own height, each node i maintains a height array with an entry $HN_{i,j}$ for each neighbor $j \in N_i$. Initially, the height of each neighbor is either set to NULL, $HN_{i,j} = (-, -, -, -, j)$, where j is the RID of the neighbor router, or is set to ZERO, $HN_{i,did} = (0, 0, 0, 0, did)$, where did is the destination prefix for which the algorithm is running. This latter information is known because node i maintains two auxiliary tables per interface: the RID-to-prefix table $R_{i,r}$ mapping the RID r of an adjacent router to its set of advertised network prefixes (derived from its neighbors' HELLO messages), and a prefix table P_i in which it stores the known network prefixes of any adjacent non-router nodes (as derived from ARP or some other means).

Each node i (other than the destination) also maintains a link-status array with an entry $LS_{i,j}$ for each link $(i, j) \in L$, where $j \in N_i$. The state of the links is determined by the heights H_i and $HN_{i,j}$ and is directed from the higher node to the lower node. If a neighbor j is higher than node i, the link is marked upstream (UP). If a neighbor j is lower than node i, the link is marked downstream (DN). If the neighbor's height entry, $HN_{i,j}$, is NULL, the link is marked undirected (UN). Finally, if the height of node i is NULL, then any neighbor's height which is not NULL is considered *lower*, and the corresponding link is marked downstream (DN).

When a new link $(i, j) \in L$ is established (i.e., node i has a new neighbor $j \in N_i$), node i adds entries for the new neighbor to the height and link-status arrays. If the new neighbor is the destination, the height entry is set to ZERO, $HN_{i,did} = (0, 0, 0, 0, did)$; otherwise it is set to NULL, $HN_{i,j} = (-, -, -, -, j)$. The corresponding link-status, $LS_{i,j}$, is set as outlined above.

Proactively-Optimized TORA

When reacting to link failures, a TORA DAG may become less optimally[3] directed and some of its δ_i's may lose their distance significance. A recent enhancement to the protocol is the ability to periodically propagate **optimization** (OPT) packets outwards from the destination, reception of which resets the reference levels of all nodes to zero and restores distance significance to their δ_i's. Among the fields carried in an OPT are a destination network prefix, a sequence number and a δ value. An OPT propagates throughout the domain as a directed flood installing or overwriting heights in the routers for the destination (a DAG such as that depicted in Figure 4 results). A δ value received from a node j implicitly advertises a height of $(0, 0, 0, \delta, j)$. A node receiving an OPT updates its notion of the transmitting neighbor's height. A node i which has already processed the OPT (as determined by examination of the sequence number) will then silently discard the packet. Otherwise, it will update its height to $(0, 0, 0, \delta+1, i)$ and retransmit the packet. Should a network portion be cleared of height information following a failure, a node requiring a route may initiate on-demand route construction with a QRY as in normal TORA processing.

In addition to serving as a routing enhancement, the optimization process helps ensure that router state errors—resulting from undetectable errors in packet transmissions or other sources—do not persist for arbitrary lengths of time. Any router state which is not explicitly refreshed by the optimization process will eventually time-out and be deleted (i.e., returned to a VOID value). Thus, the periodic optimization also serves as soft-state confirmation of route validity. This optimization/refresh process permits introduction of far-reaching control message propagation as a secondary, background mechanism that is independent of network topology dynamics. Although the overhead associated with this optimization mechanism grows proportionally to the square of the number of destinations, the periodic interval is controllable. Ideally, this procedure should occur at a low rate, to minimize the impact of the added overhead.

3.1.2 Construction and Maintenance of BS/NAP DAGs

Within the EMA, each BS/NAP must advertise a network prefix into the FMC domain so that

(i) packets can be default routed to hosts with addresses covered by these prefixes and

(ii) a virtual, bi-directional link exists between any pair of cooperating BSs for their communication during mobile handover.

We accomplish this by having a separate version of TORA build and maintain a DAG for each prefix. Initial DAG construction occurs by having each BS/NAP router, on initialization, transmit an OPT packet into the domain. The BS/NAP DAGs are then maintained via a combination of the aforementioned mechanism and normal TORA reactive processing.

---

[3] "Optimality" in this paper refers to shortest-path routing based on a distance measure, typically hop count in IP networks, and not on delay minimization, throughput maximization or other criteria oftentimes synonymous with optimal routing.

3.2 Mobile Host Routing

In the EMA each host is allocated an interface address covered by the allocating BS or NAP network prefix. While the host is "at home", packets are routed to the host via this network prefix. If the host moves away from its allocating BS, host-specific state is injected in the network during handover to overrule (via longest prefix match forwarding) the host's default DAG and redirect packets to the hosts current location.

Numerous mechanisms can be defined to achieve this. Given that the inter-BS routing algorithm we have chosen is TORA, we seek a mechanism that operates in harmony with TORA's notion of height-based routing, and permits a large degree of flexibility concerning the method and scope of host-specific state injection.

At this early stage of work, the design objective is to *localize* the scope of handover-induced messaging so as to reduce the processing load on routers as much as possible while maintaining routing efficiency. Domain scalability is the end goal.

3.2.1 TORA Handover Processing

FMC TORA differentiates nodes into two classes: routers and hosts. Routers execute the full FMC protocol while hosts execute only a limited state machine that does not involve packet forwarding. Base stations (BS) are routers, and mobile hosts (MH) are handed over between routers. In general, routers may also be mobile (e.g. mobile ad hoc networks), but we will not consider that case here.

The original TORA protocol operates reactively, both in terms of initial route construction and route maintenance in response to *unforeseen* topological changes. We have already seen how TORA's route construction process can be made more proactive. Now, we will show how TORA can respond to *known* topological changes, whether foreseen or unforeseen but anticipated.

In certain wireless technologies (e.g. GSM), handovers can be *predicted* based on signal-to-interference measurements at nearby BSs. After the appropriate handover criteria are reached, a handover procedure can be initiated from an OBS to a NBS in response to a *known* topological change. In other instances, e.g. with technologies not supporting handover prediction, the *unanticipated* loss of a link should not be immediately interpreted at the OBS as an undesirable link failure. Instead, the OBS should *wait* for a time to see if the MH reappears, connected either to itself or to a NBS. If it appears at a NBS, the OBS again treats this as a *known* topological change and reacts accordingly. If it does not appear, the OBS eventually declares link failure and reacts accordingly.

We now present a general mechanism for moving a destination identifier from one node to another in a TORA domain. We will then specify that to the case of handover processing.

Transference of Destination Identifiers

TORA nodes may be associated with multiple destination identifiers, and a separate DAG may be built for each identifier. It is possible to change the association of a destination identifier from one node to another while preserving routing integrity by *lowering* the destination's height. For example, when a DAG is initially constructed for a destination identifier *did*, it is assumed to have a ZERO height (0, 0, 0, 0, *did*), as shown in Figure 5a which depicts the transfer of a host destination identifier. To transfer the destination, its height is lowered to (-1, 0, 0, 0, *did*) as shown in Figure 5b. In essence a new *negative* reference level (-1, 0, 0) is defined for the destination. A **temporary virtual link** to the new node (node *k*) is built at the old node (node *h*) so that the DAG remains well defined with only a single destination. Otherwise, the TORA algorithm at the old node should have to react since it would have lost its last downstream link to the destination. Note that the routing is still loop-free. However, for the old node *j* to forward packets to destination *did*, it has, in effect, to *tunnel* them to the new node *k*, which would then forward them to the destination. The new node *k* accepting the destination assumes a height (-1, 0, 0, 1, *k*), indicating that it is one hop from (and still *higher* than) the destination.

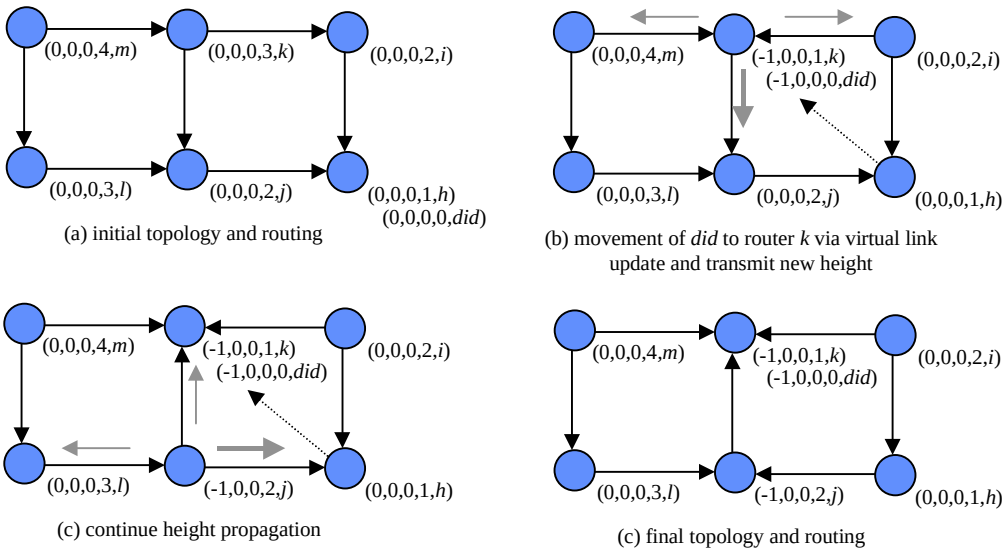


Figure 5: Movement of a host destination identifier between routers

What remains now is to remove the virtual link. This can be accomplished by sending an **unicast-directed update** (UDU) message to the old node *h* re-directing a portion of the DAG. In the example, node *k* generates an UDU and sends it towards node *h* via node *j*. The UDU carries a copy of the destination identifier (*did*), the transmitting node's height for the *did*, (-1, 0, 0, 1, *k*), a destination identifier for the old node (not shown), and a designated next-hop receiver (*j*). Note that a copy of the UDU should also be sent to any other neighbor of node *k* that already knows its height (nodes *i* and *m* in the example) to inform them of the height change. But only node *j*, being the designated receiver of the UDU, may continue forwarding it towards the old node. The designated receiver *j* adjusts its height by adopting the reference level of the UDU and increasing its δ value by one before forwarding. It also picks another designated receiver, node *h* itself in this case. On receipt of the UDU, the old node will remove the virtual link. This process can be repeated as the identifier is moved from node to node, with the τ value being decreased by one each time.

The preceding example showed the movement of a host interface identifier associated with a router—in effect, the movement of an attached host from one router to another. In a similar fashion, it is also possible to move a destination identifier (a network prefix) associated with one router's interface to another router's interface (this permits subnet mobility). The visibility of the destination identifier's new location can be increased by more broadly advertising the new location on handover. In the limit, one could advertise a new reference level throughout the entire domain, effectively performing a re-optimization of the DAG for the most negative reference level. There is a trade-off between routing efficiency and advertisement scope, and the trade-off is topology dependent.

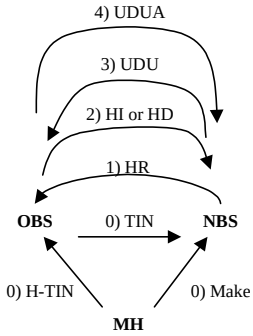


Figure 6: Basic EMA Handover Messaging

EMA Handover Messaging

If MH movement is predicted, then the OBS may be informed by the MH (if mobile assisted operation is implemented) with a **host tunnel initiation** (H-TIN) packet or via a layer 2 signal. This causes the OBS to build a temporary, soft-state tunnel towards the NBS and to send a **tunnel initiation** (TIN) packet to the NBS. This message may give the NBS advance warning of handover. The tunnel can serve to help avoid packet loss during any link dead-time. This sequence of events and the tunnel's construction are optional. What is *not* optional is the construction of a virtual link at the OBS. If handover is predicted, this virtual link is accompanied by a tunnel and is terminated at the NBS. If handover is not predicted and the link to the MH is suddenly lost, a virtual link to the MH itself is retained for some time while the OBS awaits notification of the MH's location.

Otherwise, the EMA handover model has its focus at the NBS and all mandatory messaging begins there. On arrival at the NBS, the MH (operating in mobile-assist mode) brings up a new link for IP purposes (i.e. **Make**) to the NBS. This triggers the NBS to send a **handover request** (HR) message to the OBS. The OBS responds with either a

handover initiation (HI) (i.e. AAA information and associated state) packet if handover is permitted, or else with a **handover denial** (HD) packet. The HR packet is repeatedly sent until either a HI or HD is received, or it is determined that the OBS is unreachable. If handover is permitted, the HI packet begins a three-way handshake to transfer control of the mobile to the NBS. On receipt of the HI, the NBS initiates routing redirection by sending an UDU towards the OBS. This is sent reliably hop-by-hop towards the OBS, and may be resent multiple times until an **UDU acknowledgement** (UDUA) is received at the NBS or the OBS is determined to be unreachable. This message exchange remains the same for both BBM and MBB handover, whether or not the handover can be predicted.

"Anticipated" Break-Before-Make Processing

We illustrate the preceding message exchange with a "GSM-like" scenario of BBM handover in Figure 7.

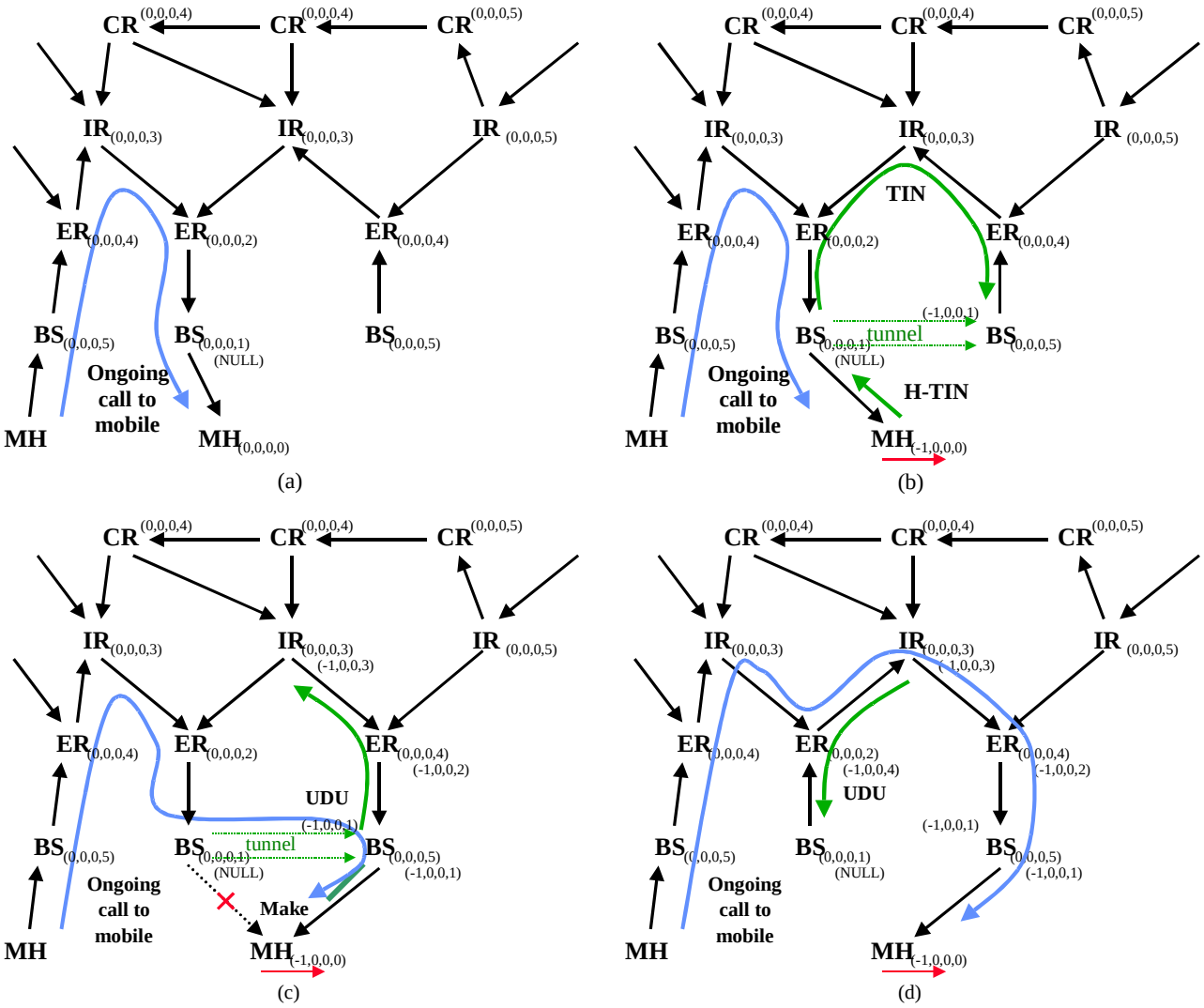


Figure 7: Anticipated BBM handover

We can only show a portion of the message exchange due to space constraints. The H-TIN packet initiates tunnel creation and transmission of the TIN packet. The Make event is seen in Figure 7c, which initiates the UDU (we skip the HR and HI phases here) to redirect routing via injection of host-specific routing state (shown next to the OBS prefix DAG state). Meanwhile, packets have been tunneled towards the NBS since the Break event. Packet flow is redirected in Fig. 7d after the UDU hits the crossover router, and the tunnel comes down when the UDU hits the OBS. We also omit the UDUA phase due to space limitations.

As the mobile migrates, a similar sequence will be repeated at each handover. Hopefully the reader will be able to construct similar message diagrams for the other cases involving unanticipated BBM, anticipated MBB and unanticipated MBB from what we have presented. These handover forms may occur as the mobile moves between different types of layer 2 technologies (e.g. GSM to Bluetooth handover) generating different handover event sequences.

Recall, it will be necessary to re-allocate the IP address back to the ABS after a sequence of handovers. This is accomplished via a sequence of messages very similar to the handover processing (only in reverse) where the IP address is handed back to the ABS and all host-specific state is erased from the network which we cannot show due to space limitations.

4. Performance

Our initial performance analysis has focused on assessing the potential scalability and routing efficiency of the proposed routing algorithm. Opnet-based simulations have been conducted on hierarchical mesh topologies of various sizes (see Table 1) consisting of **core routers** (CR), **intermediate routers** (IR), **edge routers** (ER), BSs and MHs. In all topologies the CRs are connected at the top level in a full mesh (i.e. a clique), the IRs are either single or dual (D)-homed to the CRs, the ERs are likewise single or dual-homed to the IRs (or CRs according to the topology), while the BSs are always single-homed to the ERs. Link speeds are set to 155 Mbps between fixed nodes, and the wireless transmission rate was set to 4 Mbps. Figure 8 gives a rough representation of a dual-homed topology.

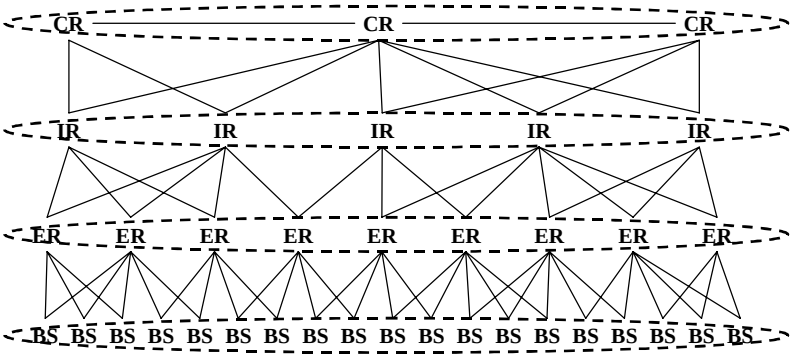


Figure 8: An example dual-homed topology

We assume all users in the system are mobile. Each BS controls a block of 200 IP addresses for allocation to the MHs on call establishment. The number of MHs in the system is equal to 200 times the number of BSs in the topology. A cellular telephony traffic pattern is assumed. Mobiles move and establish calls to each other according to GSM call statistics, with an average cell dwell time of 87 seconds and an average call length of 131 seconds, both drawn from exponential distributions. Simulation run times were 10000 seconds. A stressing mobility model is assumed. Mobiles pick a random direction and move in a straight line until hitting the edge of the domain, as which time they pick another random direction and continue moving, etc., until the call terminates.

	CR2_ER4_BS16	CR4_ER16_BS144	CR4_ER40_BS400	CR4_IR16_ER160_BS1600
CR	2	4	4	4
IR	0	0	0	16
ER	4	16	40	160
BS	16	144	400	1600

Table 1: Dimensions of simulated domain topologies

The objective of the localized handover model is to keep host-specific state out of the core and as far towards the network edge as possible. Straight-line movement maximizes stress on the network as the furthest topological distance is crossed in a given call, requiring the deepest host route injection into the core.

4.1 Storage Costs

Due to hierarchical aggregation effects, movement within a hierarchical mesh topology tends to concentrate the effects of host route injection at routers deeper in the core. Our initial focus is to discover the extent to which the localized handover model injects host routes into the core, as this would not only indicate the storage burden on routers, but would also allow us to estimate the message processing load on the routers. Figure 9 shows the percentage of host route insertion into route caches (computed as a percentage of the total number of MHs in the system) as a function of router type (i.e. core depth) for four single-homed topologies. Interestingly, these percentages improve (i.e. decrease) as the topology scales upwards for all levels in the hierarchy, although the actual number of routes does increase with increased domain size as also shown in Figure 9.

Core networks are commonly dual-homed for robustness against link and router failures. The effect of dual-homing the IRs and ERs in the largest topology is seen in Figure 10. With dual homing all host routes vanish from the CRs and are effectively redistributed to the much larger number of ERs. The benefits of localization become more pronounced as the network becomes more mesh-like (or less tree-like) near the edges. The handover message processing and state installation remains further out towards the edge. From a population of 1600*200=320000 mobile hosts, the average number of host routes in the IRs was 11,616, and the maximum number never exceeded 18,200 routes at any IR at any point in the simulation.

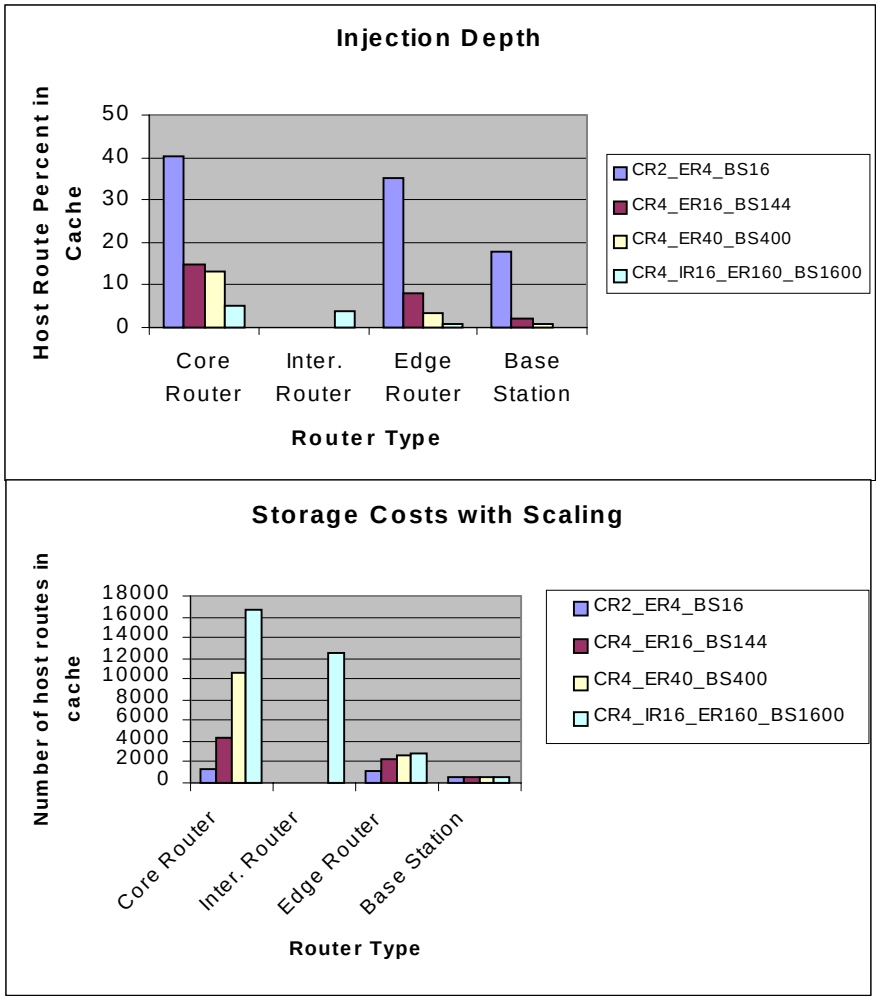


Figure 9: Injection depth as a function of hierarchical level

4.2 Routing Efficiency

In this initial study we are also interested in quantifying the routing efficiency of the proposed algorithm. TORA forgoes support of a shortest-path computation to achieve localization. When TORA constructs a DAG with an OPT packet, the packet floods throughout the network and the resulting DAG depends entirely on the order of OPT transmissions and receptions in the network. Afterwards, each node has an estimate of the distance (i.e. the hop count) to the destination through each downstream neighbor, known from that neighbor's δ value. In these simulations, TORA implements a **lowest-neighbor** forwarding policy, meaning that a node forwards packets to the destination via its neighbor with the *lowest* height value for that destination. We conjecture that forwarding over a DAG using this policy will roughly approximate shortest-path routing.

To test this conjecture, we compare the proposed approach (FMC TORA) with a hybrid algorithm (TORA/ILS) where TORA host-specific DAGs are overlaid atop an Ideal Link State (ILS) routing algorithm. ILS routing is used to compute shortest-paths between BSs, and provides a benchmark for measuring the non-optimality of the TORA DAGs. The comparisons must be made over dual-homed topologies, as the single-homed topologies are essentially trees rooted at a clique of CRs and offer no path diversity. The comparative results (see Figure 11) show that the degree of non-optimality is negligible for the simulated topologies. We are as yet unable to obtain comparative data for the largest topology due to the complexity of executing Dijkstra shortest-path computations in very large topologies; such work is underway. The results are not surprising as the DAGs have not undergone link reversals and remain as constructed.

Node movement provides added potential for route degradation. It is likely that the routing towards the mobile will become less optimal over time as a mobile moves through a mesh topology away from the ANAS. To avoid this effect, the optimum mobile routing approach would be to utilize an ILS algorithm for mobility management, whereby each handover consists of a link failure at the OBS and a link activation at the NBS, with this information being flooded throughout the network for each handover. All routers could then maintain up-to-date shortest paths to the mobile. Unfortunately, this approach is infeasible for all but the smallest domains. Nevertheless, we simulated against an Omniscient Link State (OLS) approach—one which always knew the mobiles' locations without requiring signalling—to see how far FMC TORA deviates from this ideal. The results can also be seen in Figure 11. Here, we

see meaningful differences in routing efficiency of 0.18%, 4.8% and 3.1% in the 16, 144 and 400 BS topologies, respectively. The performance variations seen here do not yet indicate a trend (this is not surprising given the compositional variation in the simulated topologies) and we are not yet able to determine whether the negative effect of mobility on routing efficiency decreases as network size increases. It does appear from the data gathered thus far that mobility has a greater impact on routing efficiency than does non-optimal DAG formation. Our conjecture is that through a proper combination of topology design and signalling we can keep the mobility-induced degradation within acceptable limits, even in large topologies where the routing is oftentimes stretched across many cells. A sufficiently hierarchical topology creates route aggregation that reduces the number of potential preferred paths between distant BSs. The results thus far indicate that there is sufficient hierarchical aggregation remaining in the dual-homed topologies to limit the degradation to less than 5% with highly localized signalling.

5. Alternative Approaches for IP Mobility Support

5.1 Mobile IP

Mobile IP [MobileIP] is a well-known technique for supporting edge mobility through the use of stateful intelligence and the permanent use of tunnels. Mobile IP is used in our approach as the only credible means to support hand-over between Autonomous Systems whilst trying to preserve the current IP interface address and dependent IP sessions. Mobile IP effectively takes the position that IP routing should not inherently support IP interface movement due to the implications on the scaling of the intra-domain routing. It is therefore deemed better to add functionality to hide the interface movement from the routing protocol(s).

The authors of this paper concur with this position for traditional routing protocols such as OSPF, where the consequence of mobility in any reasonable domain is clearly disastrous for routing state and message overhead. However, we believe that the proposed approach can achieve sufficient scaling to be commercially useful, and the case for Mobile IP against a routing solution becomes a more detailed comparison of interactions with other IP/cellular protocol features, system reliability, system overhead, time to market and ultimately cost, etc. We think that with suitable routing technology, the Mobile IP solution will in many cases be inappropriate.

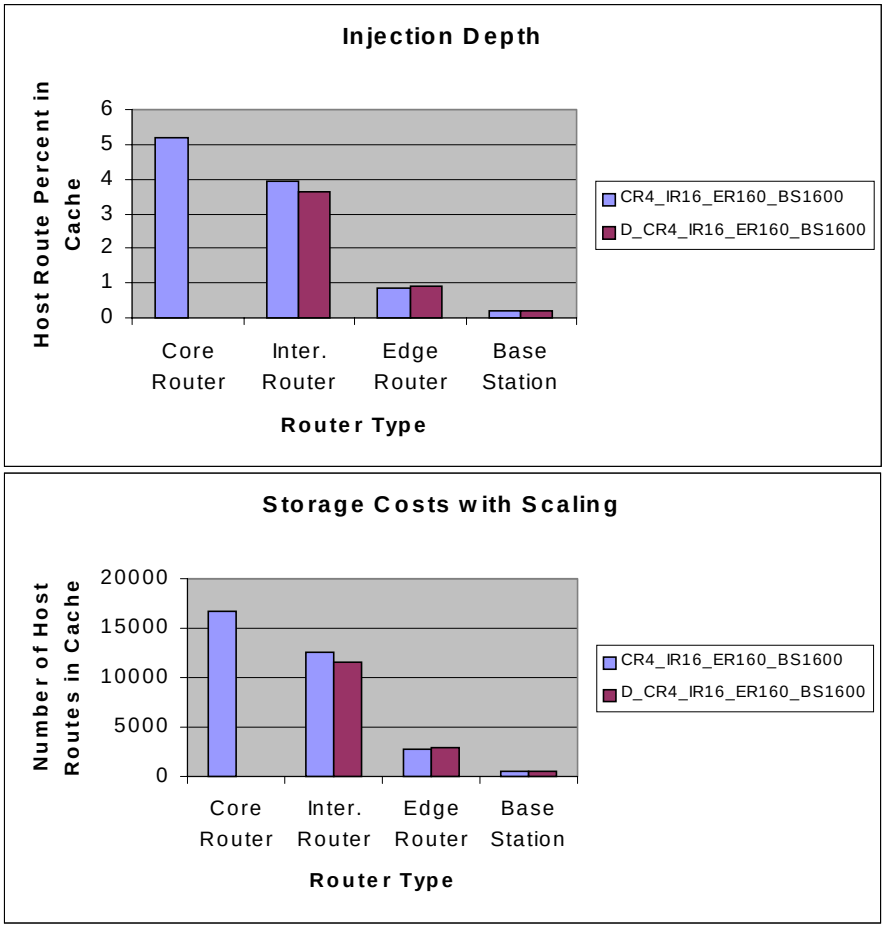


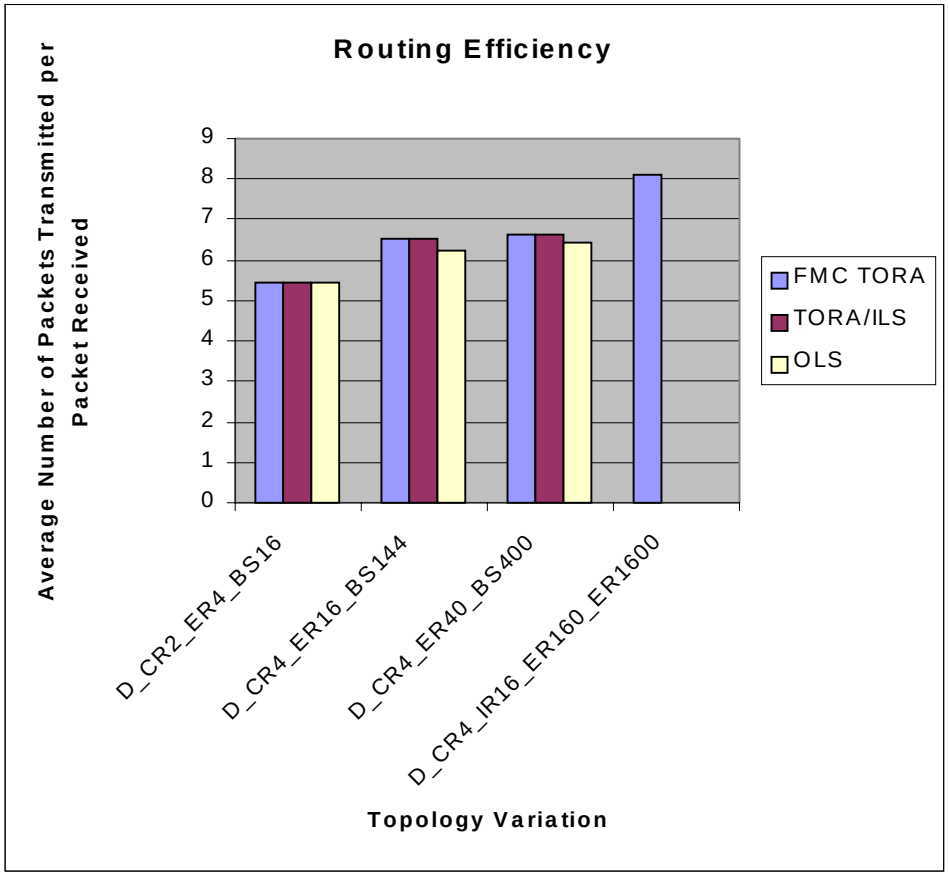
Figure 10: Effect of dual-homing on injection depth

5.2 Cellular IP/HAWAII

Cellular IP [CellularIP, CellularIPdraft] is an existing proposal which fits some aspects of the EMA; in particular in that it uses Mobile IP inter-domain and advocates use of a constant in-session address. It provides for handover-based redirection and soft state-based maintenance of host-specific routing and paging entries. These entries point to a central domain router, and the redirections modify a set of default routes collectively forming a tree. HAWAII [HAWAIIdraft] is a proposal similar to Cellular IP. It also advocates the use of a constant in-session address and

Mobile IP inter-domain. It also provides for handover-based redirection of host-specific routing paths rooted at a central core router.

Cellular IP and HAWAII differ from the proposed architecture in their use of a central routing tree. In the proposed approach, host routes modify a distributed, prefix-routed mesh topology and form route sets other than trees leading to reduced configuration, greater resilience, shorter data path lengths and topological design freedom. In addition, these approaches appear not to make provision for the temporary tunnelling of in-flight packets whilst redirect routing converges, which can lead to data packet loss depending on topology, convergence time, link speeds, control packet loss recovery times and traffic load.



F

Figure 11: Routing degradation due to non-optimal DAGs and mobility

More fundamentally, the proposed approach is a full routing algorithm, employing hard state for both host-specific and prefix-based forwarding entries, rather than a soft-state overlay technique for mobile hosts independent of the underlying fixed routing. Trust is placed in the host-specific entries, and periodic soft-state reconfirmation is not utilized[4]. The proposed system is designed to scale. Central to this design objective is the limitation of network control signalling, with the understanding that control message processing is a principal bottleneck to system scalability. Frequent route verification may potentially overload routers in the domain sizes we envision.

Future work will, in part, address system robustness. The underlying TORA algorithm for inter-BS routing—originally a MANET protocol—is already designed to handle arbitrary link and node failures. At any point in time the mobile host-specific entries also form a TORA DAG, and TORA's failure reaction mechanisms could be used to clean up state after failures. However, we are developing a separate self-healing mechanism for the host-specific routing entries.

5.3 On-Board Switch Experiments

Recently the U.S. Government demonstrated a mobile networking system wherein airborne routers formed a small MANET in the sky. The routers were networked with link state routing. Each router also had a high power WaveLAN interface with which it communicated with numerous land-based mobile hosts. In effect, each airborne router was a mobile BS which had a link to each of its associated mobile hosts. Due to the combination of host and router mobility, it became necessary to handover mobiles between routers. This was accomplished via link state-based flooding of the mobile's new topological location. This approach has limited scalability as mentioned previously, but it is interesting as it demonstrated that a fully routed, link state-based solution could be used for mobility management.

[4] While the BS DAGs are periodically re-confirmed, this is done primarily to re-optimize routing. It is not anticipated that re-optimization will occur frequently enough to serve as an effective means of soft-state route maintenance, and this aspect is merely a by-product.

6. Conclusions

We have presented an approach for FMC domain-based routing that treats fixed and mobile-terminating traffic in the same fashion. It has the obvious advantage that many IP features designed for the fixed network will also work for mobile hosts. For example, the approach is amenable to traffic engineering. By avoiding the use of long-term tunneling, separate flows terminating at a mobile are visible and may be handled separately according to their traffic classes as part of a DiffServ-based quality of service architecture. The approach also has the potential to replace or obviate the need for many layer 2 mobility signalling technologies, as well as to replace some existing IP routing protocols in fixed domains.

The work is admittedly at an early stage, and the simulation studies are far from complete. But concept combines three previously separate technology spaces—fixed routing, cellular mobility and MANET routing—and points in a potentially promising direction.

The approach hinges on the use of flat, non-shortest path routing algorithm in a large-scale, hierarchical mesh topology. This fusion of topology and algorithm capitalizes on the strengths of the routing algorithm while diminishing its weaknesses. For example, the use of a non-shortest path algorithm can lead to inefficient routing. However, a sufficiently hierarchical topology reduces the number of potential paths effectively limiting the extent to which routing can diverge from optimal. At the same time, a sufficient degree of meshing within the topology permits exploitation of the algorithm's localization properties thereby avoiding far-reaching control message propagation. Freed from this convergence bottleneck, the topology can be made larger, potentially much larger than is possible with traditional shortest-path routing technology. It is apparent then that obtaining the best performance requires a trade-off—a proper balance of topology design and control during handover to fully exploit the algorithm's localization properties while preserving routing efficiency.

Our future work will focus in several areas. The current handover model is highly localized, involving only the MH, the OBS and NBS and the set of routers on the preferred paths between OBS and NBS; in this sense similar to the handover model of HAWAII. Localization is desirable for scalability in that control messaging is minimized. However, there are costs associated with this degree of localization. As the mobile continues moving, it will leave a trail of host-specific state in the network. Also, in certain topologies, the routing may still be less optimal than desired due to node mobility. Additional handover mechanisms are under consideration as performance enhancements to address these issues, and significant simulation study is needed. We will also develop the mechanisms needed to handle dynamic topologies for the edge mobility enhancements to TORA, and plan to put forth a companion paging architecture.

References

[CellularIP] A. Valko, ``Cellular IP: A New Approach to Internet Host Mobility,'' ACM Computer Communication Review, Vol. 29, No. 1, January 1999.

[CellularIPdraft] A. Campbell, J. Gomez, C-Y. Wan, Z. Turanyi and A. Valko, "Cellular IP", *Internet-Draft (work in progress)*, draft-valko-cellularip-01.txt, October 1999.

[EMAdraft] A. O'Neill, M. S. Corson, "Edge Mobility Architecture", *Internet-Draft (work in progress)*, draft-ietf-oneill-ema-00.txt, October 1999.

[GPS] NAVSTAR GPS user equipment introduction, MZ10298.001 (February 1991).

[HAWAIIdraft] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan and L. Salgarelli, ``IP micro-mobility support using HAWAII,'' *Internet-Draft (work in progress)*, draft-ietf-mobileip-hawaii-00.txt, June 1999.

[IAB99] M. Kaat, "Overview of 1999 IAB Network Layer Workshop", *Internet-Draft*, draft-ietf-iab-ntwlyrws-over-01.txt, Oct. 1999.

[MobileIP] C.E. Perkins, ``IP Mobility Support,'' Internet RFC 2002, Oct 1996.

[NTP] D. Mills, "Network time protocol, specification, implementation and analysis", *Internet RFC-1119* (September 1989).

[TORA] V. Park, M. S. Corson, "The Temporally-Ordered Routing Algorithm", *Proc. IEEE INFOCOM '97*, Kobe, Japan, April 1997.

[TORAdraft] V. Park, M. S. Corson, "Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification", *Internet-Draft (work in progress)*, draft-ietf-manet-tora-spec-02.txt, October 1999.

[TORAthesis] V. Park, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks", *Masters Thesis*, University of Maryland, College Park, Maryland, 1997.