A MODEL OF SEAMLESS IP MOBILITY FOR FUTURE WIRELESS ACCESS NETWORKS

Theo Pagtzis and Peter Kirstein

Department of Computer Science
University College London
Gower Str. WC1E 6BT
London U.K.
{t.pagtzis,p.kirstein}@cs.ucl.ac.uk

ABSTRACT

This paper presents an architecture and protocol in support of *seamless mobility* for future IP Radio Access Networks (IP-RANs). It encompasses a novel approach for seamless handoff and proactive relocation of *IP roaming state*. The latter establishes a generic substrate for proactive *state relocation* of different context classes relating to the state of IP connectivity for a mobile node (MN).

To address such form of IP mobility, the proposed model identifies a *tentative mobility matrix* (TMM), which represents an accurate mapping between a *mobility neighbourhood vector* (MNV), surrounding the current point of attachment of an MN and the correct underlying *routing neighbourhood vector* (RNV), over arbitrary routing topologies.

Sustained IP connectivity is achieved by introducing a *1-neighbour-lookahead (1-NL)* view of IP roaming state derived from the established TMM component; seamlessness is pursued through mapping of the 1-NL component to some *proactive care-of address* (PCoA) onto the IP Multicast domain; this allows abstracting a plurality of candidate care-of address instantiations of the MN onto a single handoff routing identifier.

Proactive Mobility, Seamlessness, Cell bounce effects, ping-pong effects, context transfer, state relocation, Proactive care-of address, multicast, mobility neighbourhood, routing neighbourhood, IP roaming state, tentative mobility matrix.

1. INTRODUCTION

The advances of wireless IP networking technologies [1, 2, 3, 4, 5] and portable computing terminals [6, 7, 8] are reaching a stage of maturity, where users expect convergence in the wireless/wired IP network infrastructure, enabling **true** diverse access capabilities: access *on the move*, global span, **constant connectivity**, uniform performance characteristics, seamlessness, IP transparency.

In this engendered paradigm shift in traditional access practices manifested as *mobile networking*, users in command of portable wireless computing devices require access to some packet-switched, **all-IP** wireless network infrastructure. This is independent of their physical location, while moving to their ignorance over multiple coverage areas that span geographically towards some destination.

The above reasons for the departure from the circuit-switched model of personal communication systems (PCS) such as GSM [9], towards all-IP Radio Access Networks (IP-RANs) [10, 11]. In IP-RANs the mobility of a host translates to attachment on different last-hop wireless links realised as IP *cells*, independent of the underlying wireless technology, be it 802.11b/a [12, 13, 14], GPRS [15], or EDGE/UMTS [16, 17].

Coupled with the notion of ubiquitous computing [18] and *nomadic* communications [19, 20], mobile networking practices enable tangible new possibilities for novel kinds of multimedia applications 'on the move': navigation [21, 22, 23], personal locator services [24] interactive audio/video [25, 5], network games [26].

Real-time dissemination of multimedia information becomes now even more important than ever, as mobile devices and users integrate information retrieval as a peripheral task of their main activity (driving, operating, pursuing, walking, or generally 'acting'). These activities require bounded latencies if communications are to sustain real-time guarantees in terms of both acted task performance and supporting communicated information. It has been shown extensively in [27, 28, 29, 30], that for one-way delays in excess of 150 msec[1] the quality of interactive audio/video traffic degrades significantly while beyond 200 ms it is rendered unacceptable [31][2].

Towards ubiquitous mobile networking, Perkins [33] proposed extensions to protocol considerations for network layer host mobility, originally devised by Ioannidis and Maguire [34], known as Mobile IP. Posed as the dominant standard for mobile networking, Mobile IP effects a transparent mapping between the home IP address of a mobile node (CN) and a care-of IP address (CoA) acquired at the visited point of attachment; it is characterized as a *reactive* IP mobility protocol since IP connectivity provisions at a visited link are *initiated* upon detection of an incoming MN.

Despite its wide acceptance, Mobile IP has been shown [35], to be insufficient for support of real-time IP traffic. In IPv4, Mobile IP [36, 37] restricts the MN in changing points of attachment not faster than once every 1000 ms. Over IPv6 networks, Mobile IP [38], continues to lack of support for delay-sensitive IP traffic, due to network layer switching latencies incurred either by core IPv6 protocol functions or due to external factors impacting directly its reactive character.

With respect to core IPv6 protocol functions, Finney and Scott [39] verify such deficiency by showing that irrespective of the IPv6 router advertisement interval, the allocation of an IP address requires a minimum of 160 ms with no DAD hits (which can worsen latency although very rarely); such delay period is accounted from

[1] more accurately around 200 msec

[2] For a detailed elaboration over latency requirements, the reader is invited to peruse our investigations in [32]

the moment that the IPv6 stack of the MN[3] is notified for stateless address autoconfiguration until the moment that a binding update is transmitted. The above imply that allocation/activation of IPv6 addressing state generates by itself enough latency to place any active IPv6 flow on the boundaries of acceptable guarantees for real-time traffic delivery.

It is noted that, IP **roaming** (addressing) state may be only one of the types of context required to admit an MN and its active IP flows into a visited network; for instance, security context may have to be re-established prior to any packet flow; in the case of resource reservation for the purposes of Integrated Services QoS provisioning, the entire end-to-end path needs to be re-established at the new network link. For AAA admission control, credential verification must be effected with the home network prior to admission of the MN at the new network link. Each of the these context states requires one or more round trip times in terms of protocol interactions before they are established in a serial[4] manner; this is clearly beyond the control of core IPv6 or mobility protocols. The above attest that re-establishment of context state beyond IP addressing augments further the total latency incurred as the MN transits between network links so as to degrade real-time delivery guarrantees well beyond acceptable boundaries.

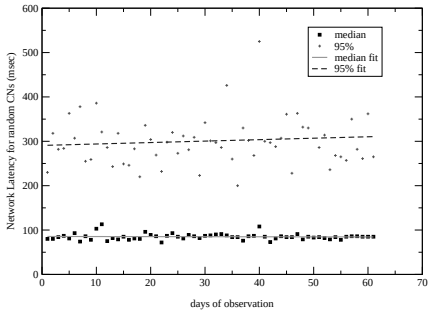

Figure 1: Network latency for random CNs

Context relocation together with other latency factors [40] constitute what we term as **latency externalities** with respect to IP mobility. Figures 1 and 2 show an indicative set of real-world network[5] measurements of round trip times from popular or random http servers acting as correspondent nodes (CNs) connecting with stationary MNs.

The latency factor over mobility has been further considered in the light of congestion by the work of Mukkamalla and Raman [41]; their results ranged between 70 ms for no congestion and 1600 ms during congestion with average values around 540 ms. For no back-off during losses the latency reached as high as 1600 ms while for lost replies peaked at 1950 ms. The work also pointed out that such figures amount also to the capability of the HA to tunnel traffic for up to 2500 hosts; latency begins to soar above this figure. Considering the capabilities cellular networks to tackle handoffs at the rate of 3000/sec, the figure becomes representative of the capacity for a single home agent and for packet size up to 250 bytes. Smaller packet size can raise the capacity of the HA.

From the above it becomes clear that *reactive* IP mobility mechanisms are expected to encounter latencies that will unavoidably impede adherence to guarantees for real-time delivery of IP traffic; the problem is bound to be further exacerbated as the MN transits at high speeds while crossing small cells[6]. For example, for a crossing rate of 33.3 m/sec[7] and a minimum overlap coverage ratio of 0.1 (100 meters) [42, 43], the MN has 2.1-3 seconds to effect the handoff assuming no BER.

To this end, we establish a model that promotes the *proactivity* in IP mobility mechanisms over future IPv6 Radio Access Networks. Our mobility model argues that IP mobility management cannot not rely on the reactivity of the upcoming visited network when real-time delivery/transmission guarantees must be assured/sustained for active IP flows to/from a visiting MN. This is far too slow as soon as the node begins to consider higher and less deterministic mobility patterns; Instead, we propose that it is the network either current or previous that must *proactively* manage and distribute a mobile node's IP connectivity (or other context) state much in advance of the MN's handoff transition.

The rest of this paper is structured as follows: Section 2 provides with some terminology and assumptions with respect to the proposed IP mobility model. Section 3 presents the proactive mobility model. Section 4 describes the mechanism for management of the mapping between the mobility and routing in Proactive IP mobility. Section 5 presents the mechanism for brokering and relocating IP roaming state to the MN. Section 6 elaborates on the abstraction of the MN's routing identifier in the IP multicast domain; section 7 describes the behaviour of the MN over Proactive IP mobility. Section 8 presents related work and a brief discussion on marked differences with the proposed mobility model. We conclude with a summary of the proposed seamless mobility model and future work in Section 9.

2. TERMINOLOGY AND ASSUMPTIONS

Our proactive mobility model assumes for simplicity an **Access Router (AR)** to be controlling a single **Access Point (AP)**, identifying a unique coverage area[8]. It is however, possible to consider

[3] which is orthogonal to any hardware optimizations at the access router

[4] one protocol after the other

[5] the raw data have been statistically analysed after kind permission from Telcordia Technologies

[6] of nominal range equal to 1 km

[7] equivalent to a vehicular speed of 165 Km/h

[8] A coverage area may be modeled as hexagonal for the purposes of adjacency and continuous coverage

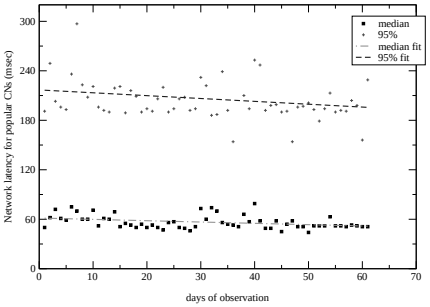

Figure 2: Network latency for popular CNs

control of multiple APs[9] from a single AR. We define an Access Point as a link layer entity that operates, currently, transparently from the perspective of IP layer. We consider an AR and AP as separate functional entities and argue that such separation is essential for future IP-RAN since it allows both *routing* as well as *coverage area diversity* [44]. This is not possible when these entities are integrated in a single physical device.

Bandwidth resource are expected to be constrained over wireless links; for this reason our model emphasizes on **minimizing signaling dependencies** on the MN. Such dependencies become more critical under harsh propagation effects which are prone to increased bit error rates (BER) during MN's movement.

The proposed mobility model is concerned only with standard cross-link network transitions without distinguishing inter- or intra-domain movement.

Each AR is assumed to transit between three possible states: `NEW`, `CURRENT` and `PREVIOUS`. Details on the transitions between different states for both the MN and the AR can be found [44].

The mobility pattern of the MN may affect the stability of the state transition process for each AR. The most representative of these is what we call *cell-bounce*. This is also known as *ping-pong* effect [45].

3. PROACTIVE MOBILITY MODEL

Contrary to the traditional reactive mobility practices, the notion of *proactivity* is considered from the perspective of the **mobility-enabled** fixed network, extended by the last hop AR through its corresponding wireless AP. It comprizes of a mobility-aware **routing neighbourhood** communicating proactively **IP roaming state** that is effective over its corresponding **mobility neighbourhood**. **IP Roaming state** is defined as the IP addressing/registration state provided to an MN at some CA within a mobility neighbourhood to maintain constant network connectivity during its mobility pattern within that neighbourhood.

A *mobility neighbourhood* is defined as the set of *geographically-adjacent*[10] coverage areas (CAs); the CA where the MN resides temporaly is identified as the *current* CA while the surrounding CA are identified as *neighbouring CAs*. A fundamental property of a mobility neighbourhood is that for each current CA there exist a set of neighbouring CAs such that all CA neighbours are immediately reachable by the current CA; this is illustrated in Figure 3.

As each CA/AP is assumed to correspond to some AR, the set of all ARs corresponding to the constituent CAs of the mobility neighbourhood is defined as a *routing neighbourhood*, depicted in Figure 3.

To distinguish between proactive mobility and other types of IP mobility management we identify three abstract types of transition establishment with respect to an AR that may accommodate an MN in the immediate future; these types facilitate the notion of a **candidate access router (CAR)** effected within a mobility neighbourhood. The three transition types are defined as:

- *reactive transition* : transition is established in response to the *reaction* of the new AR detecting an incoming MN on-link, through IPv6 neighbour discovery mechanisms [46].
- *forward-reactive transition* : transition effected as a result of a forward hint from some network entity, primarily the current AR, to the new AR. The current AR is effectively **pushing** some context state, such that the new AR is somewhat prepared about the upcoming MN visitor. This identifies an **informed reaction** on the part of the new AR. Alternatively, hint *requests* may be sent to the current AR such that some context state is **pulled** by the new AR from the current one. The transition is still effected in response to a reaction, potentially faster.
- *proactive transition*: transition administered entirely on the initiative taken from the *candidate AR(s)* to provide sufficient IP routing information that is utilized by the current AR for establishing IP roaming state **well in advance** of the handoff transition to one of these ARs by the MN; such state is self-contained and sufficient to enable an MN transition with no need for reaction on the part of the candidate AR.

The proposed model allows **complete** as well as **partial** proactive mobility management. This is because Proactive IP mobility may not be supported on *every* AR within the routing neighbourhood. In such event, the IP roaming state provided to the MN represents only part of the mobility neighbourhood; if the MN transits to an AR that does not support the proactive IP mobility, the IP Roaming state **fades gracefully** for that AR while the MN can revert transparently back to reactive IP mobility mechanisms such as [47].

3.1. Components of the proactivity mobility model

A mobility neighbourhood is represented by its **mobility neighbourhood vector (MNV)** denoted as:

$$MNV_n = \{CA_1, CA_2, \ldots, CA_n\} \quad (1)$$

where n is the number of CAs within that neighbourhood reachable from the current CA_0. In a fractal fashion, each CA within some MNV_n maintains its own mobility neighbourhood. Similarly, the underlying routing neighbourhood of the aforementioned MNV vector with respect to the CURRENT AR defines the corre-

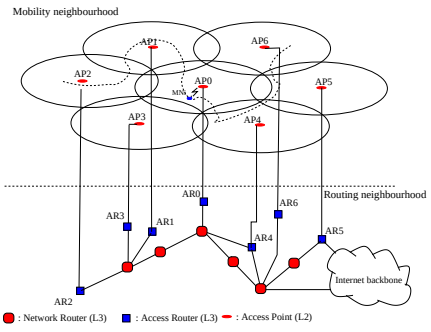

Figure 3: Mobility neighbourhood and its underlying routing neighbourhood

[9] adjacent or not

[10] with some minimal overlap

sponding **routing neighbourhood vector (RNV)** denoted as:

$$RNV_n = \{(IP, PrefLen, LLA)_{AR1}, (IP, PrefLen, LLA)_{AR2}, \vdots , (IP, PrefLen, LLA)_{ARn}\} \quad (2)$$

where IP identifies the IP address, $PrefLen$ the length of the AR's prefix and LLA the link layer address of the AR in a tuple representing a unique *RNV element*. Each AR maintains its own RNV as a one-to-one mapping with the respective MNV of the CA currently visited by an MN.

The proposed model identifies the notion of proactive **mobility-routing state** exchanged between ARs and tracked by the availability of a MNV vector that is effected over the corresponding routing neighbourhood. It comprizes of the RNV vectors from every AR mapping to some CA within a single MNV. It is, thus, effected **omni-directionaly**, i.e. 360°-wide for the travel direction of a mobile host. Mobility routing state can provide **next-mobility-hop** routing information within a mobility neighbourhood, while the corresponding ARs are **not** necessarily adjacent to each other, in the underlying network topology.

4. MOBILITY NEIGHBOURHOOD DISCOVERY

It is important to consider the mapping between the MNV vector and the underlying topology of the respective RNV vector; the significance lies on the amount signaling for discovery and subsequent configuration of AR members of the RNV vector for some MNV.

For simplicity in the model description, members of the MNV vector are assumed to be **homogeneous**; each CA provides omni-directional coverage, modelled as a hexagonal approximation of circular shape with constant diameter; this implies an MNV *adjacency pattern* of Figure 3. The proposed model is not restricted by the above modeling assumption, since its core requirement pertains to the existence of overlap between coverage areas, not their shape.

Hop-adjacency between the ARs in the RNV vector of the CURRENT AR, however, depends highly on the topology of the constituents. For this reason the model assumes *spanning-tree* topologies over the mobility neighbourhood as shown in Figure 3; that is, movement of the MN between CAs **does not** assume a direct link between their respective ARs.

The MNV and its respective RNV for fixed infrastructure networks is not expected to change often. Thus, discovery of the **MNV-RNV** mapping may be achieved as follows:

- *static configuration*: the MNV-RNV mapping is manually configured; while trivial, this approach is not scalable in terms of CA/AR deployment or failures as well as cost of ownership for the network provider.
- *dynamic learning and configuration*: employ incremental dynamic learning on each AR from information conveyed proactively by the MN. This type of learning relies on the temporal existence of bypassing MNs. Dynamic learning of this type fits naturally to the movement of MNs, since they explicitly discover adjacent CAs in transit towards a destination.

To effect the latter, each AR maintains a pre-configured **Coverage Area Tuple (CAT)**, denoted as:

$$CAT_{CA_i} = \{(l_i, L_i, r_i) : CA_i \epsilon MNV_{CA_k}\} \quad (3)$$

where l_i the latitude position of CA_i, (L_i) it longitude and r_i its radius. Such information is assumed to be available during installation of the AP.

4.1. Incremental RNV Acquisition for an AR

As a mobile host travels geographically towards some destination, it 'meets' along its path new CAs **in succession**, as shown in the mobility neighbouhood part of Figure 3; that is, each CA visited, is **adjacent**[11] with respect to the previous one and vice versa. For instance, (CA_{AP2}, CA_{AP1}) and (CA_{AP1}, CA_{AP0}) are overlapping[12] neighbours in the travel path of MN.

As the MN transits between adjacent CAs it is bound to receive new router advertisements from the respective new ARs; The 'discovery' of adjacent CAs through receipt of router advertisments can trigger by the MN an exchange of mobility routing state between the respective PREVIOUS and NEW ARs to update their RNV vector. The temporal initiation of such exchange is conditional and depends on whether the PREVIOUS AR provided previously the MN with any IP roaming state for the NEW AR 'discovered'.

Initially each AR has no mobility routing state; there exists no RNV mapping for the mobility neighbourhood of its respective CA. At that stage, the CURRENT AR cannot provide the MN with any IP roaming state; As such Proactive IP mobility cannot be effected between the MN and NEW AR; the mobility model reverts to base IPv6 mobility until mobility routing state between the two ARs has been established.

On receipt of a router advertisment, the MN checks its router prefix against any existing IP roaming state. If there is no match within its **IP roaming state Cache**, the MN, upon obtaining IP connectivity, provides the NEW AR with a unicast indirect[13] **RNV Update** message; this message is assumed be authenticated for the purposes of malicious MNs that attempt to provide bogus RNV Updates. The RNV-Update message contains the IPv6 and link layer address for the network interface of the PREVIOUS AR that effects a potentially adjacent CA as well as the CAT of that coverage area.

On receipt of the indirect RNV Update, the NEW AR first acknowledges the RNV Update to the MN by means of an **RNV Ack**. It then calculates the **adjacency factor** d_o with respect to its own CA and the CA of the PREVIOUS AR, notified by the incoming MN, as follows:

$$\begin{aligned} k_1 &= d_l - r_1 \\ k_2 &= d_l - r_2 \\ d_o &= d_l - (k_1 + k_2) \end{aligned} \quad (4)$$

[11] we consider the term neighbouring and adjacent to maintain identical semantics and as such be used interchangeably

[12] assuming overlap between coverage ares

[13] The message is considered indirect because it does come from another AR

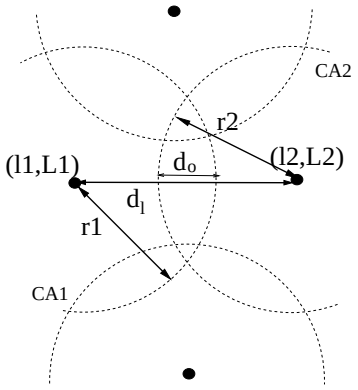

Figure 4: adjacency and overlap calculation

The distance between the APs[14], shown in Figure 4, may be expressed as:

$$d_l = R\dot{d}_a \wedge \quad (5)$$
$$where\ R\ = 6371.23\ km$$

$$d_a = 2asin(min(1, \sqrt{sin(\frac{\delta l}{2})^2 + cos(l_i)cos(l_{i+1})sin(\frac{\delta L}{2})^2})) \quad (6)$$
$$where\ \delta L = L_{i+1} - L_i \wedge$$
$$where\ \delta l = l_{i+1} - l_i$$

Assuming spherical[15], d_a has been optimized by means of half angles to effect higher accuracy for small distances between APs. The value of d_o signifies the following:

$$d_o = \begin{cases} d_o > 0 & ,CA_P, CA_N \text{ adjacent with overlap } d_o \\ d_o = 0 & ,CA_P, CA_N \text{ adjacent, but no overlap} \\ d_o < 0 & ,CA_P, CA_N \text{ potentially non-adjacent} \end{cases}$$

The receipt of the indirect RNV Update by the NEW AR, signifies that the PREVIOUS AR is currently not aware that their CAs are neighbouring. If the value of the computed $d_o >= 0$, the NEW AR stores both the IP and link layer AR address of the previous AR in its RNV vector within its **RNV Cache** and then sends a new unicast **RNV Update**, to the PREVIOUS AR; since there is adjacency between the previous and new CA, the NEW AR need only include in the message its own IPv6 and link layer address so that the PREVIOUS AR updates immediately[16] its RNV vector. On receipt of the message the PREVIOUS AR simply responds with an RNV Ack message after updating its RNV vector . Now, both PREVIOUS and NEW AR can establish mobility routing state to be used proactively for generation of IP Roaming state in either direction of MN movement between their respective CAs, as shown in Figure 5.

In the event that some element of the RNV for a single AR becomes stale[17], that element is instantly nullified; this is because the MN cannot match the routing prefix from the received router advertisment with any of its IP Roaming state cache entries. As such Proactive IP mobility cannot be effected at the MN and thus the dynamic learning mechanism for incremental RNV acquisition is initiated.

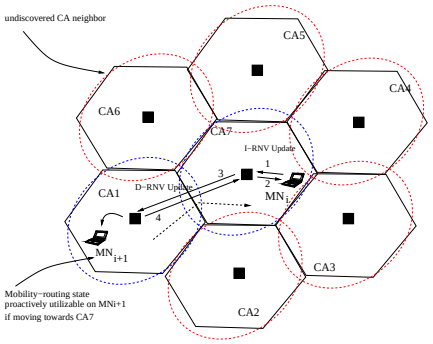

Figure 5: Partial proactive mobility through temporal reactive learning

4.2. RNV and Mobility routing state convergence per MNV

When an AR acquires the complete[18] RNV vector of its mobility neighbourhood, it then **advertises** that vector -**RNV Advertisment (RNV-Adv)**- to the members of its RNV vector. Alternatively each AR can explicitly **solicit** the advertisment of the RNV from the vector's members through a **RNV Solicitation (RNV-Sol)**.

To effect that a new instance of neighbour discovery is introduced, called **Routing Neighbourhood Vector Discovery (RNV-D)**. The RNV-D process allows the complete mobility routing state (set of RNVs) for a single mobility neighbourhood to be propagated from the center towards the **edges** of that neighbourhood. It is interesting to observe that since **every** CA and effectively AR is found at the center of some mobility neighbourhood MNV_i, the same CA/AR can be found at the edge of some other mobility neighbourhood MNV_{i+1} which overlaps (per CA) with MNV_i, as shown in Figure 6. This guarantees that propagation of the complete mobility routing state, through individual RNV Advertisments, will achieve fast convergence within a single mobility neighbourhood. The converged mobility routing state within an AR of the routing neighbourhood comprizes the **tentative mobility matrix** (figure 7) enabling per-mobility-hop routing in that routing neighbourhood.

For simplicity, when the RNV element (primarily IP address) for AR_i is found in the RNV of some AR_{i+1}, then AR_i will be referred to as the **routing-neighbour** or simply **neighbour** of AR_{i+1} (and vice versa), for the remainder of this document.

In a manner similar to router advertisements in IPv6 [46], an RNV Advert is transmitted periodically. The transmission interval is adjusted by an exponential backoff until some maximum inter-

[14] center of two adjacent CAs.

[15] Further distance optimizations may be effected by considering an elipsoid Earth shape. However, this involves somewhat more complex calculations.

[16] no need to calculate again the adjacency factor

[17] as in the event of prefix renumbering

[18] no new RNV Updates past a convergence threshold T_c

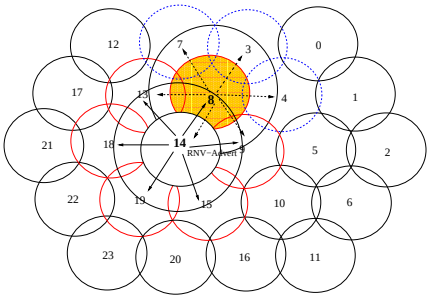

Figure 6: AR propagates its RNV towards the edges of the mobility neighbourhood

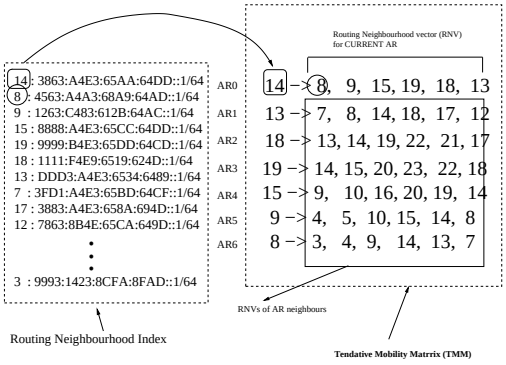

Figure 7: Mobility Routing state maintained in RNV at AR14

val threshold $Tmax_{RNVA}$. To effect responsiveness on sustained solicitations the backoff value is halved per RNV solicitation up to some minimum interval threshold $Tmin_{RNVA}$.

Periodicity of RNV Adverts is shifted by some small random delay to avoid periodic effects [48]. Solicited RNV Adverts are syncronised with the periodic transmission of RNV Adverts in line with the proposed advertisment backoff interval.

5. IP ROAMING STATE BROKERING FOR A MOBILE HOST

It is assumed that the MN establishes its home addressing state and the preferred Home Agent at its home network by means of either *remote* or *local* connection to its home network. Such connection may be effected in a manner similar to [38].

Assuming existence of home addressing state and IP connectivity through standard (Mobile) IPv6 mechanisms, the MN provides its CURRENT AR with its **link-layer address (LLA)**. On receipt, the CURRENT AR acknowledges it by sending the MN its own RNV element, while it forwards that LLA to all of its AR neighbours retrieved from its RNV cache. The MN stores the received RNV element in the event that the MN needs to trigger an indirect RNV Update to the NEW AR (no IP Roaming state for that NEW AR). The MN cannot discard this entry until either an indrect RNV Update has been acknowledged by the new AR. In the event that IP roaming state is available to the MN for the NEW AR, the cached RNV element at the MN is overwritten by the new RNV element supplied by the NEW AR.

Establishment of IP connectivity at the MN triggers a transition of the AR from the NEW to the CURRENT state. However, establishment of IP connectivity by the MN, requires that the CURRENT AR *brokers* a set of unique IPv6 unicast care-of addresses (CoA), for the MN admitted; each of these IPv6 CoAs is topologically correct in one of the CURRENT AR's neighbours; this set is termed as **soft care-of address tuple (sCoAt)**. The term **soft** implies that the CoA has been *allocated*, and may be used during and after a potential handoff transition.

The sCoA tuple consititutes the IP Roaming state for the MN since it allows roaming from the CURRENT AR in the mobility neighbourhood with proactively established signaling. This is because the candidate AR for the next handoff transition of the MN is found within that mobility neighbourhood.

The CURRENT AR unicasts the LLA of the MN to all neighbouring ARs in a **soft CoA Create** message. Each of the receiving ARs is empowered with the IP CoA generation task in a statefull fashion; as such the AR neighbour combines the LLA [49] of the MN into an IPv6 **soft CoA (sCoA)**; following generation of the sCoA, the AR neighbour performs duplicate address detection on that address[49].

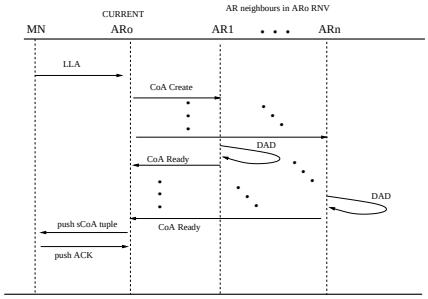

Figure 8: Message interactions for proactive IP generation

5.1. Duplication Address Detection for Proactive IP Mobility

A standard reactive DAD check under IPv6 is effectively address resolution for a tentative address[49]. This is usually performed link-locally upon generation of new IPv6 address. If a neighbour solicitation is not responded with an advertsment within some time interval the address is considered to be unique.

With the case of proactive care-of address generation, an IPv6 CoA must be generated off-link at some neighbour AR while the MN does not currently reside on that link. This is shown in Figure **??**. To detect a duplicate sCoA address, the AR neighbour first checks its neighbour cache entry for that sCoA to see if already existent. If not, it creates an entry with the sCoA and link layer address of the MN. The entry is marked with a P flag as *proactive* and set to INCOMPLETE state.

The neighbour AR then attempts DAD through standard address resolution. In particular, it sends a neighbour solicitation to the solicited-node multicast address mapping to the SCoA. It further includes as source Link Layer address option its own one on as the sender[19]. The sCoA address is found to be unique if no

[19]this function is usually performed by the MN when on-link

neighbour advertisment is received back within `RetransTime` milliseconds.

As soon as uniqueness of the soft CoA is ensured, the AR neighbour marks its the neighbour cache entry as PROACTIVELY REACHABLE; This new state in Neighbour discovery [46] is introduced for the purposes of enabling proactive reachability of that neighbour cache entry; that is to say, that the particular entry **does not require** a solicited neighbour advertisment by the MN. This is because a neighbour solicitation is used in two broad occasions:

- when neighbour reachability or duplicate address detection is effected. The last degrades to address resolution which has the same effect on-link. Here the AR is aware that there no packet has arrived that effects that neighbour solicitation.
- when a packet arrives at the AR for some host on its link, but the neighbour cache entry state has to be set to reachable. The fundamental difference with the above case is that the AR is aware that it acts in response to a packet that has arrived for the host on-link.

In both cases the neighbour solicitation requires the LLA of the MN. In each of the two cases our proposed model reacts differently. While in the first case the AR defends the soft CoA with its own LLA, in the second it simply returns the real LLA of the MN. However, the second case cannot occur in the proposed model since no HA or CN knows the soft CoA generated. This is because there has been no handoff transition yet by the MN on that AR neighbour that can trigger a Binding Update to announce the existence of the sCoA as a primary CoA for the MN.

By setting the link layer address of the MN in its neighbour cache in addition to its own LLA, the neighbour AR requires minimal information in order to activate the MN's LLA when traffic needs to be forwarded to and from the MN's particular soft CoA. This is because both the neighbour AR and the MN are configuring their neighbour cache entry in advance of transition with the particular entries marked as proactive and set in the state PROACTIVELY REACHABLE. In this manner, communication between the two entities does not require a solicited neighbour advertisment; it can effect communication of packet traffic **immediatelly**. The PROACTIVELY REACHABLE state of the cached entry is reduced to REACHABLE as soon as the MN has sent a Binding Update (BU) to its peers. The BU ensures that a stable primary CoA has been activated as detailed in following sections.

5.2. Distribution of the soft CoA

With DAD completed, each neighbour AR returns the sCoA into a unicast **soft CoA Ready (sCoA Ready)** message back to the CURRENT AR of the MN. Each sCoA received in response to the MN's LLA sent, is grouped together in a sCoA tuple comprizing part of the IP Roaming state for the MN. Note that generation of soft CoAs is distributed in the mobility neighbourhood, while the DAD function is performed on the spot at CoA generation time. Thus, no signaling needs to be *proxied* between the CURRENT AR and its AR neighbours for either CoA generation or DAD.

The CURRENT AR *injects* the IP Roaming state established, to the admitted MN, through a **Context State Push** (CS-PSH). Context in this case is IP Roaming. The message includes the number of AR neighbours providing such state, together with a unique sCoAt identifier associated with a respective **Context State Cache (CS-Cache)** entry at the CURRENT AR. In addition to the sCoA tuple, the particular context state entry includes also the following information per AR neighbour:

- the **IPv6 address** of the neighbour AR interface that effects a CA of the mobility neighbourhood. It is required to update the default router list, as default router list entries that are provisionally effected during handoff.
- the **prefix length** for that neighbour AR interface. Essential to derive link layer information for updating the MN's neihbour cache. Required to minimize or eliminate neihgbour discovery signaling when the MN handoffs to some AR neighbour.

From this information the MN can further reconstruct other routing information that complement the IP Roaming state for that mobility neighbourhood. This is:

- the AR neighbour **link layer address**. Derived by using the prefix length together with standard EUI-64 rules for interface identifier generation; alternatively it may be explicitly provided by the CURRENT AR as stored in its RNV Cache. It is used in the neighbour cache of the MN, marked with the P flag (PROACTIVE) and set in the PROACTIVELY REACHABLE state.
- the AR neighbour **routing prefix**. By using the prefix length, determine the routing prefix of the AR neighbour and populate the prefix list of the MN.

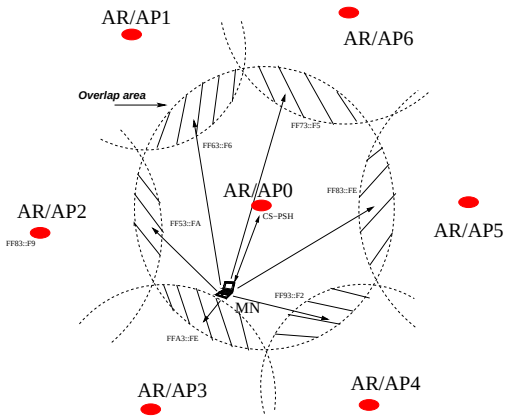

Figure 9: The proactive set of soft CoAs provides 1-domain lookahead network connectivity

It can be seen, that the proposed model does not effect sCoA tuple generation at the MN. The mobility model argues against any dependence on the response of the MN back to the CURRENT AR since it induces excessive *overhead* from proxied signaling interactions for the purposes of either address configuration, neighbour discovery, DAD or other context-specific action between the MN and AR neighbours through the CURRENT AR; proxied approaches of control signaling for IP mobility results in increased latency between the CURRENT AR and the MN, that may be significantly affected in cases of corrupted/lossy signaling due to harsh fading conditions.

6. ABSTRACTING THE MN ROUTING IDENTIFIER

Upon generation of the sCoA tuple, the CURRENT AR abstracts the tuple's constituents onto a short-lived, globally routable, unifying routing and addressing identifier, that is allocated for the MN; this is referred to as **Proactive Care of Address (PCoA)** and is *not* a unicast but a **multicast** IPv6 address. It is assumed that all ARs are multicast-enabled according to the IPv6 protocol architecture [50].

The mapping of the sCoA tuple onto a PCoA is effected *only* at an AR; it is transparent from the perspective of the peer entities of an MN. Both HA and CNs send packets towards the MN through the CURRENT AR with no knowledge about the existence of a PCoA address. By mapping the sCoA tuple over a PCoA address as shown in Figure 10, the CURRENT AR allows the MN to receive traffic through any AR neighbour in its mobility neighbourhood that is candidate for transition. This by effecting PCoA group membership reports for the individual soft CoA instantiations of the MN allocated on the respective links of the AR neighbours.

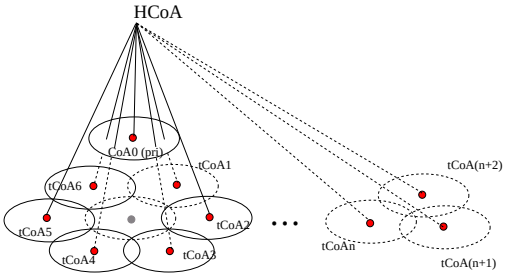

Figure 10: Mapping a multicast address to unicast IPv6 CoA listeners

Currently the PCoA address allocation process takes place only once at the first AR (home or visited) hosting the MN. The proposed mobility model assumes unique allocation of a PCoA address. Currently, this aspect is pursued by protocol recommendations from the IETF Multicast Address Allocation WG (MALLOC) [51]. Thus, an analysis on multicast address allocation is out of the scope of this paper[20].

6.1. PCoA membership management for AR neighbours

Upon allocation of the PCoA address, the CURRENT AR must inform each AR neighbour[21] participating in the sCoA tuple, to enable forwarding of traffic destined to the abstracted PCoA address in their downstream interface for which the allocated soft CoA is valid since there will be interest in that traffic by at least one host, the MN.

To inform itself the CURRENT AR solicits an **explicit 'join'** from the MN, whereas informing the AR neighbours requires the CURRENT AR transmitting an **implicit join** to the AR neighbours, according to the multicast listener discovery (MLD) standard [52, 53, 54].

[20] we investigate multicast address allocation and management issues for the purposes of this scheme in a separate paper

[21] including itself

In *explicit join*, the CURRENT AR *solicits* a membership report that must be explicitly sent by the MN to join the PCoA group. This is required to ensure that the MN configures its hardware interface for the particular PCoA address. This join solicitation is piggybacked in the CS-PSH message to the MN by means of a join-bit flag (J). The MN responds with an MLD membership report [55], that configures the multicast filter on its hardware interface [56], while the receiving AR enables multicast forwarding for that PCoA on the local link. In the case of the *implicit join*, the CURRENT AR transmits an MLD membership report to each of the AR neighbours, on behalf of the MN that may be visiting its link, since the MN is visiting the CURRENT AR or cannot be on-link with all AR neighbours; each of them, receiving an **implicit join (I-Join)** message, need also enable multicast forwarding for the PCoA address over the link where the soft CoA generated is topologically correct. Figure 11 illustrates the case where an AR neighbour (e.g AR6) has to enable group membership after receiving an implicit join solicitation from the CURRENT AR (AR0).

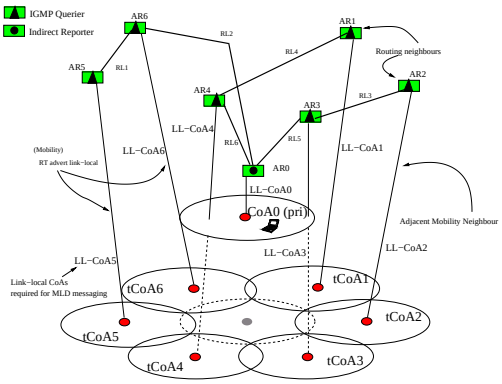

Figure 11: link-local addresses used for conventional MLD messaging

Note that group membership reporting is effected only link-locally with a multicast-enabled AR; since the MN is not physically on-link with any neighbour AR it is essential that group membership for the PCoA is managed **indirectly**.

6.2. Need for Indirect Group management

From Figure 11 it can be seen that an MLD Membership Report sent towards some AR, e.g. AR1, by the MN when on-link, normally reaches AR1 through LL-CoA1. However, in proactive IP mobility, *AR1 is an AR neighbour* with respect to the current location of the MN within the routing neighbourhood; thus, such report must now come through RL6 and then through RL4, since the MN is currently residing on-link with AR0. To achieve this, the standard multicast listener discovery mechanism must be extended to handle indirect listeners at AR neighbours within a routing neighbourhood.

In particular, on receipt of an indirect join request, the receiving AR neighbour sets an entry in its group membership list. The entry records the PCoA group and sets a timer for the membership to the `Group_Membership_interval` as per [53]. The receiving AR then generates an **indirect MLD Membership Query (I-MLD Query)** with destination address the CURRENT AR, the

originator of the indirect join. The I-MLD Query message is a *group-address-specific* query and is periodic; that is, it targets membership on a specific PCoA group and is sent by some neighbour AR querier every `Query_interval`. An I-MLD Query contains the PCoA address and a maximum response delay time within which the CURRENT AR must report back.

On receipt of an I-MLD Query, the CURRENT AR sets a delay timer adjusted according to [53]. On expiry, the CURRENT AR transmits an **Indirect MLD Membership Report (I-MLD Report)** with destination IP address the query-originator AR. This is shown in Figure 12.

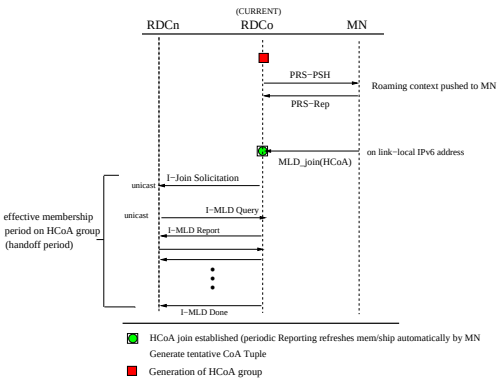

Figure 12: Periodic I-MLD messaging between ARs

Periodic I-MLD Query/Reports signal interactions refresh the timer set in the group membership list of the querier AR neighbour. They also allow for robustness in signaling since if an initial report gets lost, membership can be recovered at the next query interval. If no I-MLD Reports for the PCoA are received by the latter after expiry of response delay of the indirect query, the neighbour AR querier concludes that group membership for that PCoA must be removed from its list (link prune).

Note that the I-MLD discovery messaging is effected now through global IPv6 addresses since the indirection of the MLD signal is originating off-link with respect to the recipient AR neighbour. In addition, I-MLD messaging is exchanged only between AR entities. It must not be exchanged with a host/MN. It also requires the introduction of a P flag in the group membership list of an AR; this flag signifies a *proactive* group membership entry that is valid on-link for the mobility enabled interface of that AR.

7. MOBILE NODE BEHAVIOUR

The receipt of IP roaming state through the CS-PSH message, provides the MN with a complete mobility-aware addressing/routing 'view' (Figure 9), of the MNV vector at the CURRENT AR. Each individual soft CoA represents an instantiation of the MN in existence within the CA of an AR neighbour.

To ensure reliability in the signaling interaction with the CURRENT AR, the MN acknowledges the received CS-PSH message with a **CS Reply (CS-Rep)**. The CS-Rep message includes the number of AR neighbours received and the sequence number identifying the original CS-PSH message. If the CS-Rep message is not received within `proactive_context_push_time`, the CURRENT AR must retransmit the original CS-PSH message.

Furthermore, the MN should **accept** the provided PCoA by joining it, if the 'join' (J) and new PCoA (M) flags are set. It is possible that a new PCoA address is not provided by the CURRENT AR. In this case the MN simply (re)joins the PCoA that accepted during its last MLD membership report. Denying a provided PCoA currently implies that no proactive IP mobility is required by the MN.

Receipt of the IP roaming state requires further that the MN must:

- configure its network interface with the provided soft CoAs. The MN must transmit/receive on any of the soft CoAs **if and only if** a PCoA-Enable message is sent by it to the CURRENT AR as detailed in later section. Such messaging is effected only near a handoff transition to an AR neighbour.
- *update its default router list* with the set of AR neighbour IPv6 addresses. These addresses are expected **to be** topologically correct on the link of some AR neighbour as soon as the MN transits to it.

  Entries created must be placed at the end of default router list, marked with a proactive (P) flag. When the P flag is set, that default router entry must not be used, while the entry cannot be removed. As soon as a PCoA-Enable message is sent by the MN to the CURRENT AR, the MN may use any of the proactively marked default router entries in combination with the correct sCoA. A P flag may be removed from a defautl router list entry if the MN is on-link with the corresponding AR neighbour and a binding update has been sent to the peers of the MN about the new primary CoA. Alternatively, the MN may update its prefix list by deriving each neighbour AR network prefix through the neighbour AR IPv6 address and its prefix size. Each prefix list entry is also marked with a P flag. The rules defined for the default router list apply also for the prefix list.
- update its neighbour cache with an entry for the LLA of each neighbour ARs. Each such entry must be marked with the P flag and set to the PROACTIVELY REACHABLE (P-REACHABLE) state. Neighbour cache entries in P-REACHABLE state are excluded from checks on address resolution or neighbour unreachability detection before and during a handoff transition to an AR neighbour. A neighbour cache entry removes its P flag (while its state is reduced to REACHABLE) if and only if the corresponding soft CoA becomes the new primary CoA address for the MN.

7.1. PCoA Activation Lifetimes

Group membership for the PCoA address, requires that the CURRENT AR configures also two lifetimes L_s and L_d which are mobility management specific and extend the conventional multicast listener discovery. L_s denotes the **PCoA start lifetime** and is defined as the lifetime past which the CURRENT AR should **initiate** transmission of traffic towards the MN, through the configured PCoA group address. Its default value is -1 and denotes *infinity*; this implies that transmission of traffic over the PCoA will commence some time in the future yet undetermined.

L_d denotes the **PCoA stop lifetime** and is defined as the lifetime past which the CURRENT AR should **suspend** sending of traffic towards the MN through the PCoA address. The default

value is 0; this implies that the CURRENT AR must suspend forwarding traffic towards the MN through the PCoA group immediately (after zero time).

Suspension of traffic forwarding over the PCoA group *does not imply teardown* of the PCoA group address for the MN, at the AR. This is because traffic forwarding towards the MN through its PCoA address will be effected on a per-handoff transition basis. It further avoids additional multicast tree reconfiguration required at the CURRENT AR. Instead, as the MN moves across different AR neighbours in the mobility neighbourhood, the multicast core [57, 58] or RP [59] router effectively moves together with the movement pattern of the MN.

L_s and L_d is expected to be conditioned by the mobility vector of the MN in the case that optimal lifetimes should be pursued. The intention for these fields is to provide a time window during which handoff would be in progress as a rough estimate. More accurate lifetime refresh messages may be provided as the MN moves towards a candidate AR.

7.2. Proactive Handoff transitions in the Mobility Neighbourhood

During its movement, the MN reaches some overlap area between the CURRENT AR and one or more AR neighbours. By then, IP roaming state has been fully configured at the MN and the routing neighbourhood so that during the next handoff transition the MN does not need to configure a new CoA, but simply **activate** one of the soft CoAs as the primary one.

With respect to a handoff transition, a soft CoA may alternate between a **high secondary** and a **low secondary**; *high* refers to the set of sCoAs that are candidates for primary CoA activation; low secondary areconsidered the set of sCoAs that remain least likely to become a primary one.

Ideally, the MN needs to detect the *high secondary* soft CoA prior to a handoff transition and then inform the CURRENT AR (soon to transit to PREVIOUS state) about it. To achieve that, the proposed model introduces a mechanism for **IP disconnection avoidance (IP-DA)**, currently handled through a *pessimistic* approach.

As the MN moves away from the CA of the CURRENT AR, it crosses some overlap area between itself and a CA neighbour. In this overlap area the MN receives a different router advertisment from the one provided by the CURRENT AR; this triggers a check of the received network prefix against the existing tuple of soft CoA. The match between that prefix and a single soft CoA triggers the MN to configure this soft CoA as the high secondary CoA for the MN; in addtition the MN signals the CURRENT AR with a **PCoA-Enable (PCoA-E)** message; this message contains a soft CoA **bitfield (sCoA-B)**, a lifetime refresh for both L_s and L_d with values 0 and -1 respectively and the sCoA tuple identifier from the original CS-PSH sent to the MN. Bits marked with 1 in sCoA-B represent *high secondary* sCoAs for an MN, whereas bits marked with 0 are the *low secondary*. The sCoA tuple identifier refers to the correct PCoA associated with sCoA tuple allocated to the MN.

Enabling the correct soft CoA is essential for **upstream** transmissions from the MN to its peers. Unless the correct sCoA is enabled the MN has no means of identifying which IP address, and default route to employ in transmitting upstream during a handoff transition. What is important, however is that configuration of a topologically correct CoA is already in effect and thus there no latency induced by IP address configuration. The PCoA-E message allows the MN to receive transparently IP traffic that is sent **downstream** towards the MN.

On receipt of this message the CURRENT AR forwards any IP traffic destined for the MN by tunneling it to the PCoA address, while it stops forwarding to the primary unicast CoA of the MN. The action is *immediate* since $L_s = 0$.

Traffic sent towards the MN by peers, is now forwarded by the CURRENT AR as **encapsulated multicast payload** at the PCoA address of the MN; the current AR matches the destination address in the packet with some **Proactive Binding Cache (PB-Cache)** entry which holds, the sCoA tuple and PCoA allocated for that MN, its LLA as well as the start and stop lifetimes for that PCoA. If a match is successfull, the CURRENT AR encapsulates the received packet in a new IPv6 header with destination the PCoA group of the node; otherwise, it applies standard unicast forwarding.

Since the AR neighbours are configured to forward traffic for the particular PCoA group in their CA, traffic destined to the MN can be received over **any** IP link (CA) in the mobility neighbourhood of the CURRENT AR. This is because the configuration of the hardware interface (link-layer) multicast filter at the MN, does not depend on the unicast IP CoA allocated for the MN, but only on the last four octets of the PCoA. Thus, all is required between the MN and an AR neighbour is link-layer connectivity.

The IP disconnection avoidance mechanism, introduces the notion of a **IP multicast encapsulation** for the purposes of forwarding traffic to multiple link instantiations that the MN is probable to exist. This type of IP encapsulation requires the use of a **multicast tunnel (MT)** flag placed as a destination option in the outer IPv6 header of the encapsulating packet. The CURRENT AR must also mark the `<Next Header field>` within the encapsulating UDP header, with a special type that is called `IP_ENCAP` and denotes an encapsulated packet as the payload of a UDP header.

On receipt of the encapsulated packet, the MN must check at the IP layer that the destination options of the packet whether the MT flag has been set; in addition the MN checks the UDP header, whether the `<Next Header field>` has been set with the `IP_ENCAP` value denoting UDP encapsulation of an IP packet. If this is the value of the next header field in the UDP header then, the decapsulated packet is then re-submitted back to the IP stack. The packet would now have as destination address the on-link CoA at the CURRENT AR which the MN must sustain as an active CoA until its peers have acknowledged a standard Binding Update sent by the MN.

Traffic will be destined to the PCoA group indefinetely ($L_d = -1$); that is, until explicitly requested to suspend forwarding through the PCoA address. This is because the MN may bounce multiple times between neighbouring CAs, shown in Figure 13, causing oscillations in the signaling of PCoA activation/suspension at the CURRENT AR.

The MN continues to receive traffic over its PCoA until it attaches to some AR neighbour plus a time period T_e that intially varies between 250 and 500[22] ms. T_e is introduced for the purposes of sustaining reception of traffic through PCoA during cell bouncing effects in the MN's movement pattern between its CURRENT AR and a neighbouring AR; The time period T_e is referred to as **Extended PCoA-Rx Time**. The MN maintains also a **cell-bounce accumulator (CBA)** that tracks the number of

[22] this figure needs further experimentation

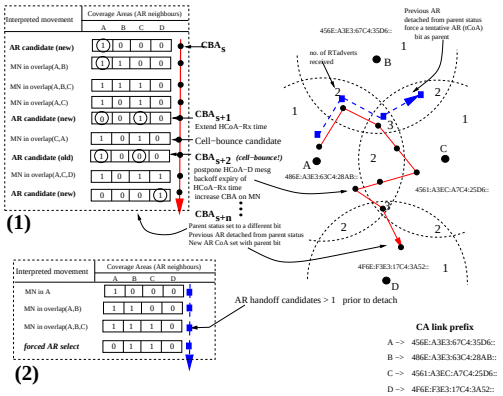

Figure 13: Bouncing effects incorporated in the movement pattern of the MN

bounces between two ARs. This accumulator geometrically increases T_e for each increment of its counter for the individual MN. The delay T_e is bounded by an upper delay maximum defined as `Max_Random_PCoA_Rx Stop_Delay` equal to 3000 ms[23]

When the extended PCoA-Rx time elapses and CBA counter has not increased, the MN sends a standard Binding Update to its peers (CNs and HA) to inform about the new, stable and topologically correct primary CoA (Figure 14). At the same time the MN sends a **PCoA-Disable (PCoA-D)** message to the PREVIOUS AR, through its new primary CoA, to request suspension of traffic forwarding through the PCoA. The PCoA-D message contains the MN's PCoA address together with refreshed lifetimes for $L_s = -1$ and $L_{d=0}$.

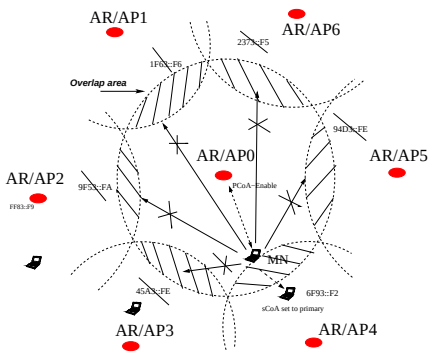

Figure 14: MN activates the correct soft CoA

The PCoA-D message instructs further the PREVIOUS AR to manage group membership of the PCoA address; it includes 'pruning' of neighbour AR 'traffic listeners'[24] that do not belong in the MNV-RNV mapping of the new CURRENT AR. The PCoA-D message includes also the sCoA bitfield with the new primary (ex soft) set.

[23] initial and max value may be conditioned by the size of the overlap area between two CAs and the speed of the mobile

[24] PCoA group forwarders

7.3. sCoA tracking and Cell-Bounce accumulator (CBA)

The CBA operates on the sCoA bitfield maintained by the MN. In this bitfield, the CBA tracks the CURRENT AR by assigning a **parent** status to its primary CoA bit. Any bit within the sCoA bitfield is set to 1 when the MN matches the network prefix in a received *IPv6 router advertisment (RT-advert)*, with the network identifier in some allocated soft CoA from its sCoA tuple. The parent status is reset to the new primary CoA when a Binding Acknowledgement is received by the first of the communicating peers of the MN.

Transition over some overlap (CA) area between the CURRENT AR and some AR neighbour(s), causes the MN to receive one or more new RT-advert messages originating from that AR neighbour(s). Receipt of such messages trigger a flip of the respective sCoA bit to 1 within the bitfield. Similarly, lack of receipt of RT-advert messages trigger a flip of the respective sCoA bit to 0 within the sCoA bitfield.

When the number of received RT-adverts (Rx_{RT_advert}) is reduced to one, the corresponding AR neighbour becomes the *immediate* handoff candidate. At this stage the CBA compares the state of the sCoA bitfield for the new AR candidate CBA_{s+1}, with the bitfield state of the previous AR CBA_s. An equilateral bit flip between CURRENT AR and an AR neighbour (i.e. $CBA_{s+1} \neq CBA_s$) signifies a potentially **clean** handoff from the PREVIOUS AR.

However, the non-determinism of the mobility pattern of the MN does not allow a clear distinction between a *clean* handoff and an *imminent* cell bounce. For this reason, forwarding over the PCoA address is **sustained** for time T_e. If within this time T_e a different (new or old) router advertisment is received then a **cell-bounce candidate** is established; it is, however, considered only a candidate since it is not certain whether the MN will indeed move back to the PREVIOUS AR or it is simply stretching movement in the overlap area.

Movement oscillation between two CAs (i.e. cell-bounce) is considered to be **imminent** iff it occurs within the interval T_e and manifests itself by means of a bitfield state $CBA_{s+2} = CBA_s$. In this case, the CBA triggers a binary geometric backoff in the expiry of T_e with a corresonding increase of this time period. CBA increases its counter and continues to track changes in the bitfield state by iterarting the process.

When a *potential* clean handoff sustains for T_e with no increase in the CBA counter, it is considered to be a **definite** clean handoff; this triggers a PCoA-D message to be sent by the MN towards the PREVIOUS AR. There is no handoff effected for a cell-bounce between the CURRENT AR and any of its neighbours in the event that the MN returns back to the CA of the CURRENT AR; it is only required to send the PCoA-D message to the CURRENT AR such that the latter can suspend the forwarding over the PCoA group.

In a definite clean handoff, **before** sending the PCoA-D message to the PREVIOUS AR, the MN must:

- configure its link-local address that is valid over the new link. Effect the correct primary CoA.
- send a neighbour advertisment to remove P flag from the neighbour cache of the NEW AR.
- send a Binding Update to its peers; at the same time set the parent flag to the new primary CoA in its sCoA bitfield.
- reset CBA counter to 0 and continue parent tracking of the new CURRENT AR.

7.4. Refreshing the soft CoA tuple

As soon as the MN has moved over the link of an AR neighbour, it is essential that it maintains a **valid** sCoA tuple; all of its allocated CoAs must be valid within the *new* mobility neighbourhood of the new CURRENT AR.

To effect that, the MN must request from the new CURRENT AR to provide the MN with a **sCoA tuple Update (sCoAt-Update)**; that would include soft CoAs from the **new** AR neighbours, valid in the RNV of the new CURRENT AR, together with a bitfield that invalidates **reduntant** soft CoAs in the RNV of the new CURRENT AR. Furthermore, the PREVIOUS AR should receive an invalidation message about the redunant soft CoAs and AR neighbours that maintain group membership in MN's PCoA group address. It is noted that the new CURRENT AR is aware of the LLA of the MN since it created a soft CoA for that MN during movement over the PREVIOUS AR.

For instance, in the MNV depicted in Figure 15, the first sCoAt allocation would require six (6) new soft CoAs. At the next handoff the soft CoA tuple need only be refreshed with another 3 soft CoAs. This is half the number of soft CoAs originally allocated.

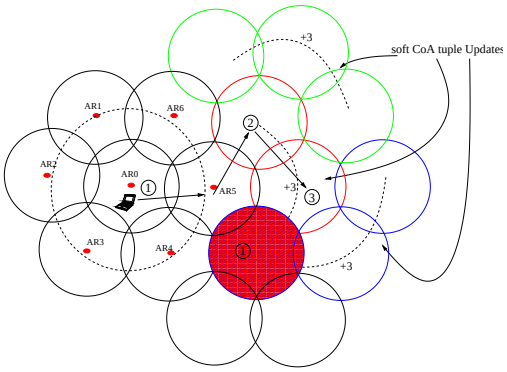

Figure 15: CoA resuse within the SCoA tuple during continuous movement of the MN

The efficiency of the above mechanism depends on how the PREVIOUS AR resolves the identity of **true** redundant ARs and their respective soft CoA.

7.5. Resolution of redundant SCoA AR neighbours

As soon as the MN has sent a Binding update towards its peers, it is considered to have **settled** within the new CURRENT AR. Upon settlement, the MN sends a **sCoAt Refresh Request (sCoAt-RR)** message to the new CURRENT AR; the message includes also the address of the PREVIOUS AR. This implies that the MN need not receive completely new IP roaming state, since it does not provide its link layer address; a refresh of its IP roaming state is only required.

On receipt of this message the NEW AR determines the AR neighbours **common** to both NEW and PREVIOUS ARs, by checking the address of the PREVIOUS AR against the contents of its RNV-Cache shown in Figures 16 and 17. By determining the common ARs between with the PREVIOUS ARs, the CURRENT AR can initiate generation of **fresh** IP Roaming state from the new AR neighbours as described in section 5. In addition it **includes** the new AR neighbours as 'listeners' to the existing PCoA group address of the MN.

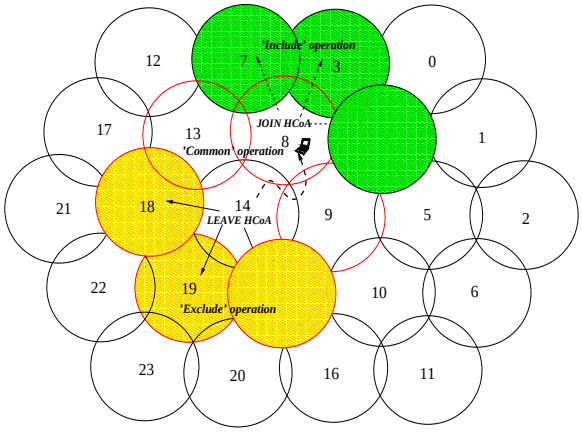

Figure 16: Sustaining accurate mapping of AR neighbours on MN's PCoA group routing identifier

In a similar fashion, the NEW AR sends an sCoAt-RR message to the PREVIOUS AR; on receipt, the latter **excludes** 'redundant' ARs from PCoA group membership. This is achieved by suppressing the sending of I-MLD Reports to the 'redundant' ARs neighbours; in addition, the PREVIOUS AR sends indirect neighbour advertisments to these AR neighbours such that the latter can remove neighbour cache entries that pertain to the IP roaming association with the MN.

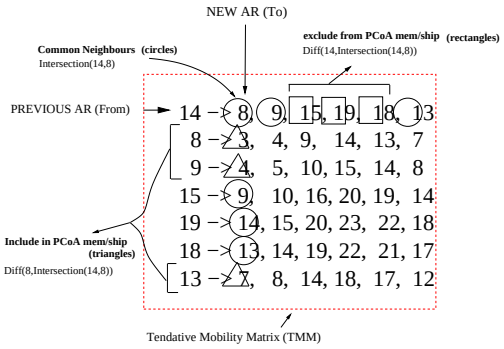

Figure 17: Resolving the AR participants that need to be included and excluded from the PCoA group

The sCoA Update calculation operations, namely *common, include, exclude* can be performed independently by both AR, either PREVIOUS or NEW. This is more important for the NEW AR as it needs to provide the MN with an **sCoAt delta** that includes the new soft CoA as well as an indication for the soft CoA that must be removed from the MN's sCoA tuple in its IP Roaming context. The The sCoA Update operations are defined as follows:

$$Common(rnv_p, rnv_n) = \bigcap(rnv_p, rnv_n)$$
$$Include(rnv_p, rnv_n) = rnv_n - Common(rnv_p, rnv_n)$$
$$Exclude(rnv_p, rnv_n) = rnv_p - Common(rnv_p, rnv_n)$$

where $rnv_{p,n}$ are the RNV vectors of PREVIOUS and NEW ARs maintained in their RNV Cache. Figure 17 illustrates the mobility traffic matrix maintained in the RNV-Cache of an AR; according to the above operations, the resulting vectors V_{common}, $V_{include}$ and $V_{exclude}$ take the following sample values according to Figure 16:

$$rnv_p = rnv_{14} = \{14, 8, 9, 15, 18, 19, 13\}$$
$$rnv_n = rnv_8 = \{8, 3, 4, 9, 14, 13, 7\}$$
$$V_{Common(rnv_{14}, rnv_8)} = \{8, 14, 9, 13\}$$
$$V_{Include((rnv_{14}, rnv_8)} = \{3, 4, 7\}$$
$$V_{Exclude((rnv_{14}, rnv_8)} = \{15, 18, 19\}$$

7.6. Forced disruption of IP connectivity

It is possible that due to limitations in wireless coverage, the movement pattern of a MN may enforce a disruption of IP connectivity in its current mobility neighbourhood; typical example is movement through road tunnels, underground train stations. In these cases, the MN appears from the perspective of the network as unreachable; it is also possible that the MN may attempt to resume IP connectivity at some AR out of the mobility neighbourhood of the CURRENT AR.

In the event of a forced disruption of IP connectivity, the CURRENT AR delays the expiry of any state maintained in its Context state Cache for time T_{tdi}; this time period is called **transient disconnection interval (TDI)** and represents the interval permissible for transient IP connectivity disruptions before the MN is considered to be off-link or simply disconnected by the AR. Upon expiry of this time interval, the CURRENT AR expires the corresponding entry in its CS-Cache as well as ensures that all AR neighbours remove their membership for the particular PCoA group address of the MN.

Upon re-connection with an expired TDI at the same CURRENT AR the MN must re-aquired new IP Roaming state for the mobility neighbourhood of the CURRENT AR. If the TDI interval has not expired the MN simply refreshes its neigbour cache entry for the CURRENT AR as REACHABLE according to [46].

In the event that the MN re-connects with an AR different from the CURRENT one, the MN must discard the IP Roaming state stored and provide its link layer address in the new CURRENT for the acquisition of fresh state.

8. RELATED WORK AND DISCUSSION

Recently, several protocols have been proposed in support of IP mobility for next generation (IPv6) radio access networks in support of seamless mobility. One of these is the fast handoffs [45] proposal; it is similar to [44] approach with respect to the concept of pre-registration of the MN with other ARs that may be candidates for handoff. Our proactive mobility model is significantly different from the fast handoffs proposal for a number of reasons. Our mobility model does not employ proxy neighbour discovery messaging since tasks like duplicate address detection mandated by IPv6 Neighbour Discovery standard, require additional signaling which also induces delays of at least one RTT between the MN and *every* candidate AR.

Furthermore, our model caters for a candidate access router discovery algorithm. The fast handoffs proposal has no such mechanism. This is an essential part for a seamless IP mobility model. In addition, our model provides a signaling substrate for proactive context transfers from AR neighbours; the fast handoffs proposal is explicitly designed towards seamless handoffs only. What's more, the fast handoff proposal lacks of robustness with respect to cell-bouncing effects (a.k.a ping-pong); this is because it assumes existence of multiple tunnels towards the MN from candidate ARs. On the contrary, our scheme employs only a single tunnel which is effected over multicast; this ensure that the MN can *freely* bounce between any AR within the mobility neighbourhood.

Alternative approaches in Mobile IP have also been proposed in [60], [61], [62]. These schemes employ IP-Multicast for addressing and forwarding of packets to MNs on an end-to-end basis; i.e. the source is destined at either the CN or the HA while MN is the receiver for the purposes of minimal latency handoffs. To this end a small group 'multicast' solution has been proposed [63] and [64]. Both schemes utilize principles from [65] and they employ a multiple unicast destination option at the routing header of a packet. This notion is similar to the route segments of [66] since both schemes rely on unicast routing to deliver the packet. The schemes however, rely on the provision of all CoA destinations to the peer entities from the MN.

The proactive mobility model is architecturaly different from all the above schemes. In the majority of the mobility proposals, IP multicast is employed either end-to-end or as a pseudo multicast mechanism where the protocol abstracts many unicast destinations as a source at the CN or HA. In the proposed mobility model IP multicast is employed **locally** with respect to the location of the mobile node (i.e. mobility neighbourhood). As such, none of the above schemes proposes a mechanism of establishing such mobility neighbourhood. Furthermore, there is no consideration in any of the proposed mobility mechanisms for state relocation with respect to multiple mobility contexts. Our abstraction of IP Roaming state as a representative class of relocatable state context allows to populate the Context State cache with multiple contexts/capabilities that neighbour ARs may need to provide to the CURRENT AR towards the seamless movement of the MN in its mobility neighbourhood.

9. CONCLUSIONS AND FUTURE WORK

This paper presented a novel model for proactive IP mobility with respect to seamless transitions between different points of attachment. We have argued that configuring addressing state in reaction to a handoff transition as effected by current mobile IPv6 mechanisms and protocols is not sufficient for satisfying seamless IP connectivity in the light of real-time IP traffic delivery or transmission to/from mobile nodes. We are currently working on a full scale simulation model of the proposed mobility model with full compliance over IPv6 protocol specifications. Results from these simulations are currently work in progress and will be the objective of a following paper.

10. ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their helpful comments that improved the presentation of the paper.

11. REFERENCES

[1] H. Bolcskel, A.J. Paulraj, K.V.S. Hari, R.U. Nabar, and W.W. Lu, "Fixed Broadband Wireless Access: State of the Art, Challenges, and Future Directions," *IEEE Personal Communications*, vol. 39, no. 1, pp. 100–108, Jan. 2001.

[2] W. Webb, "Broadband fixed wireless access as a key component of the future integrated communications environment," *IEEE Communications*, vol. 39, no. 9, pp. 115–121, Sep. 2001.

[3] A. Jamalipour and T. Tung, "The Role of Satellites in Global IT: Trends and Implications," *IEEE Personal Communications*, vol. 8, no. 3, pp. 5–11, Jun. 2001.

[4] L. Bos and S. Leroy, "Toward an all-IP-based UMTS System Architecture," *IEEE Network*, vol. 15, no. 1, pp. 36–45, Jan. 2001.

[5] J. Huber, D. Weiler, and R. Brand, "UMTS, the Mobile Multimedia Vision for IMT 2000: a Focus on Standardization," *IEEE Personal Communications*, vol. 38, pp. 129–136, Sep. 2000.

[6] A. Fasbender, F. Reichert, E. Geulen, J. Hjelm, and T. Wierlemann, "Any Network, any Terminal, Anywhere," *IEEE Personal Communications*, vol. 6, no. 2, pp. 22–30, Apr. 1999.

[7] T. Farnham, G. Clemo, R. Haines, et al., "IST-TRUST: a Perspective on the Reconfiguration of Future Mobile Terminals using Software Downloads," in *Proceedings of 11th IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sep. 2000, pp. 1054–1059.

[8] M.W. Oliphant, "The Mobile Phone meets the Internet," *IEEE Spectrum*, vol. 36, no. 8, pp. 20–28, Aug. 1999.

[9] M. Mouly and M. B. Pautet, *The GSM System for Mobile Communications*, published by the authors, 1992, ISBN 2-9507190-0-7.

[10] J. Kempf, "IP in the RAN as a Transport Option in 3rd Generation Mobile Systems," MWIF WG4: IP in the RAN Technical Report MWIF 2001.084, MWIF Forum, Apr. 2001.

[11] J. Kempf, "OpenRAN Architecture in 3rd Generation Mobile Systems," MWIF WG4: IP in the RAN Technical Report MWIF 2001.084, MWIF Forum, Sep. 2001.

[12] 802.11 WG, *"IEEE 802.11 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification Standard"*, IEEE, Oct. 1999.

[13] 802.11 WG, *IEEE 802.11b High Rate Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification Standard*, IEEE, Oct. 1999.

[14] T. Pagtzis, P.T. Kirstein, and S. Hailes, ""Operational and Fairness issues with Connection-less traffic over IEEE802.11b"," in *Proceedings of IEEE International Conference on Communications (ICC)*, Jun. 2001.

[15] 802.11 WG, *Digital Cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS); Service Description Stage 1/2*, IEEE, etsi en 301 113/v6.3.1 (2000-11) edition, Oct. 1999.

[16] 3GPP2, "3G (Parallel) Specification Section," Sep. 2001, http://www.3gpp2.org/Public_html/specs/index.cfm.

[17] 3GPP, "3G Specifications Section," Sep. 2001, http://www.3gpp.org/3G_Specs/3G_Specs.htm.

[18] R. Want, B. Schilit, N. Adams, R. Gold, K. Petersen, D. Goldbery, M. Ellis, and M. Weiser, *Mobile Computing*, chapter *The ParcTab Ubiquitous Computing Experiment*, pp. 45–101, Kluwer Publishing, Feb. 1997.

[19] T La Porta, K. Sabnani, and R. Gitlin, "Challenges for Nomadic Computing: Mobility Management and Wireless Communications," *ACM Journal in Mobile Networking and Applications*, vol. 1, no. 1, pp. 3–16, 1996.

[20] L. Kleinrock, "Nomadic Computing - an opportunity," *ACM Computer Communication Review*, vol. 25, no. 1, pp. 38–47, Jan. 1995.

[21] R. Jana, T. Johnson, S. Muthukrishnan, and T. Vitaletti, "Location-based Services in a Wireless WAN using Cellular Digital Packet Data (CDPD)," in *Proceedings of 2nd ACM International Workshop on Data Engineering for Wireless and Mobile Access*, Jul. 2001, pp. 74–80.

[22] Y. Inoue and M. Nakagawa, "MAC protocol for Inter-vehicle Communication Network using Spread-Spectrum Techniques," in *Proceedings of Vehicle Navigation and Information Systems Conference*, 1994, pp. 149–152.

[23] S. Tsugawa, Shin Kato, K. Tokuda, T. Matsui, and H. Fujii, "A cooperative driving system with automated vehicles and inter-vehicle communications in Demo 2000," in *Proceedings of Intelligent Transportation Systems*, Jul. 2001, pp. 918 –923.

[24] H. Koshima and J. Hoshen, "Personal Locator Services Emerge," *IEEE Spectrum*, vol. 37, no. 2, pp. 41–48, Feb. 2000.

[25] L. Hanzo, P. Cherriman, and E. L. Kuan, "Interactive Cellular and Cordless Video Telephony: State of the Art System Design Principles and Expected Performance," *Proceedings of the IEEE*, vol. 88, no. 9, pp. 1388 –1413, Sep. 2000.

[26] T. Henderson, "Latency and User-behaviour on a Multiplayer Game Server," in *Proceedings of 3rd International Workshop on Networked Group Communication (NGC)*, Nov. 2001, pp. to–be–published.

[27] G. Karlsson, "Quality Requirements for Multimedia Network Services," in *Proceedings of Radiovetenskap ach kommunikation*, Jun. 1996, pp. 96–100.

[28] T. Kurita, S. Iai, and N. Kitawaki, "Effects of transmission delay in audiovisual communication," *Electronics and Communications in Japan*, vol. 77, no. 3, pp. 63–74, 1995.

[29] Y. Wang, M. Claypool, and Z. Zuo, "An Empirical Study of RealVideo Performance Across the Internet," in *Proceedings of ACM SIGCOMM Internet Measurement Workshop*, Nov. 2001.

[30] B.E. Maki and W.E. Mcllroy, "The Control of Foot Placement during Compensatory Stepping Reactions: Does Speed

of Response take Precedence over Stability?," *"IEEE Transactions on Rehabilitation Engineering"*, vol. 7, no. 1, pp. 80–90, Mar. 1999.

[31] ITU-T, *Series G: Transmission system and media, digital systems and networks: One-way transmission time*, May 2000.

[32] T. Pagtzis, "Models of Seamless Mobility in IP-enabled Wireless Networks," Technical Report UCL/CS-TR-01-595, Dept. of Computer Science, University College London, London, UK, Sep. 2001.

[33] C. Perkins, "Mobile IP," *IEEE Communications*, pp. 84–99, May 1997.

[34] G. Q. Maguire Jr. J. Ioannidis, D. Duchamp, "IP-Based Protocols for Mobile Interworking," in *Proceedings of ACM SIGCOMM'91*, Sept. 1991, pp. pp. 235–245.

[35] R. Caceres and L. Iftode, "The Effects of Mobility on Reliable Transport Protocols," in *Proceedings of 14th International Conference on Distributed Computing Systems*, Jul. 1994, pp. 12–20.

[36] C. Perkins, "IP Mobility Support for IPv4 (revised)," Internet Draft, Sep. 2001, draft-ietf-mobileip-rfc2002-bis-04.txt.

[37] C. Perkins, "IP Mobility Support," RFC 2002, Internet Engineering Task Force, Oct. 1996.

[38] D. Johnson and C. Perkins, "Mobility support in IPv6 (work in progress)," Internet Draft, Internet Engineering Task Force, Nov. 1998.

[39] J. Finney and A. Scott, "Implementing Mobile IPv6 for Multimedia," in *Proceedings of GEMISIS/IEE/BCS Symposium on Multimedia Network Technology*, May 1998, vol. Digest No. G/MNT/1/1998.

[40] Telecordia Technologies, ""Internet Quality of Service Assessment"," Web page, Oct. 2001.

[41] S. Mukkamalla and B. Raman, "Scaling and Latency Issues in Mobile-IP," http://www.cs.berkeley.edu/ adj/cs294-1.s98/projects/MobileIP/sld001.htm, Apr. 1998.

[42] J. Wigard, T.T. Nielsen, P.H. Michaelsen, and P. Morgensen, "On a handover algorithm in a PCS1900/GSM/DCS1800 network," in *Proceedings of IEEE The Semi-annual Vehicular Technology Conference (VTC-Spring)*, Apr. 1999, pp. 2510 –2514.

[43] A.J.M. Ransom, "Handoff considerations in microcellular systems planning," in *Proceedings of IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Aug. 1995, pp. 804 –808.

[44] T. Pagtzis and P. Kirstein, "A Framework for Proactive Mobility in Mobile IPv6," Internet Draft, Jan. 2001, draft-pagtzis-mobileip-proactivev6-00.txt.

[45] C. Perkins G. Tsirtsis, A. Yegin, G. Dommety, and K. El-Malki, "Fast Handovers for Mobile IPv6 (work in progress)," Internet Draft, Internet Engineering Task Force, Oct. 2000.

[46] T. Narten, E. Nordmark, and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)," "RFC" 2461, Internet Engineering Task Force, Dec. 1998.

[47] C. Perkins and D. Johnson, "Mobility Support in IPv6 (work in progress)," Internet draft, Internet Engineering Task Force, Nov. 2000.

[48] S. Floyd and V. Jacobson, "The Synchronization of Periodic Routing Messages," in *IEEE/ACM Transactions on Networking*, April 1994.

[49] S. Thomson and T. Narten, "IPv6 Stateless Address Autoconfiguration," "RFC" 2462, Internet Engineering Task Force, Dec. 1998.

[50] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," RFC 2460, Internet Engineering Task Force, Dec. 1998.

[51] S. Hanna, B. Patel, and M. Shah, "Multicast Address Dynamic Client Allocation Protocol (M ADCAP)," RFC 2730, Internet Engineering Task Force, Dec. 1999.

[52] Steve Deering, Bill Fenner, Brad Cain, A. Thyagarajan, and I Kouvelas, "Internet Group Management Protocol, Version 3," Internet Draft, Mar. 2001.

[53] W. Fenner, "Internet Group Management Protocol, Version 2," RFC 2236, Internet Engineering Task Force, Nov. 1997.

[54] S.E. Deering, "Host extensions for IP multicasting," RFC 1112, Internet Engineering Task Force, Aug. 1989.

[55] S. Deering, W. Fenner, and B. Haberman, "Multicast Listener Discovery (MLD) for IPv6," RFC 2710, Internet Engineering Task Force, Oct. 1999.

[56] M. Crawford, "Transmission of IPv6 Packets over Ethernet Networks," RFC 2464, Internet Engineering Task Force, Dec. 1998.

[57] A. Ballardie, "Core Based Trees (CBT version 2) Multicast Routing," RFC 2189, Internet Engineering Task Force, Sept. 1997.

[58] D. Waitzman, C. Partridge, and S.E. Deering, "Distance Vector Multicast Routing Protocol," RFC 1075, Internet Engineering Task Force, Nov. 1988.

[59] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, and L. Wei, "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification," RFC 2117, Internet Engineering Task Force, June 1997.

[60] S. Seshan, H. Balakrishnan, and R. Katz, "Handoffs in cellular wireless networks: The Daedalus implementation and experience," 1996.

[61] A. Helmy, "A Multicast-based Protocol for IP Mobility Support," in *ACM 2nd International Workshop on Networked Group Communication (NGC2000)*, November 2000.

[62] J. Mysore and V. Bharghavan, "A New Multicasting-based Architecture for Internet Host Mobility," in *Proceedings of ACM MobiCom 97*, September 1997.

[63] Jiwoong Lee, "SGM support in Mobile IP," Internet Draft, Oct. 2000.

[64] Yutaka Ezaki and Yuji Imai, "Mobile IPv6 handoff by Explicit Multicast," Internet Draft, May 2001.

[65] R. Boivie and N. Feldman, "Small Group Multicast," Internet Draft, Feb. 2001.

[66] M. Parthasarathy A. E. Yegin and C. Williams, "Mobile ipv6 neighborhood routing for fast handoff," Internet Draft, Internet Engineering Task Force, Oct. 2000.