WiMob 2010 - 6th IEEE Intl. Conf. Wireless and Mobile Computing, Networking and Communications. Niagara Falls, Canada. Oct 2010.

A Comparative Assessment of Routing for Mobile Networks

Devan Rehunathan
University of St Andrews
School of Computer Science
Email: dr@cs.st-andrews.ac.uk

Saleem Bhatti
University of St Andrews
School of Computer Science
Email: saleem@cs.st-andrews.ac.uk

***Abstract*—Wireless mobile devices are becoming increasingly prevalent in society. As a result, aggregation of network connectivity through the use of mobile networks is becoming increasingly relevant to service providers as well as for mobile users. The current approach being pursued within the IETF Mobile Extensions for IPv6 (MEXT) WG, is based on the Network Mobility (NEMO) architecture. NEMO uses IP-in-IP tunnelling for providing mobile network capability on an existing IPv6 network. This approach can result in non-optimal routing between source and destination nodes. Other proposals such as OptiNets extend NEMO and try to address issues such as sub-optimal routing. There are alternative approaches also being proposed, such as the Identifier Locator Network Protocol (ILNPv6), which is based on the use of naming, to enable a flexible and integrated mobile network capability based on IPv6. We have conducted a comparative analysis of the cost of providing optimal routing, in terms of packet and bandwidth overhead, based on an emulation, using data from the London Circle Line metropolitan railway as a scenario. Our analysis shows that these different approaches to mobility offer significantly different performance trade-offs in routing for mobile networks, depending on the constraints of the network scenario.**

I. Introduction

As mobile devices become more ubiquitous, it becomes efficient to aggregate them together as *mobile networks*, especially when the movements of many devices are synchronised. These mobile networks can be one way of providing manageable, continued network access to a large number of mobile nodes. Mass transport scenarios such as passengers in a train who wish to remain online are one such practical application of mobile networks, as well as vehicular networks and military networks [10].

Currently the Network Mobility protocol (NEMO)[1] [3] is being used within the IETF MEXT WG as the basis of network mobility support. We consider two additional protocols in this paper; OptiNets [7] which extends NEMO and seeks to address NEMO's sub-optimal routing issues, and ILNPv6 which departs from the tunnelling model for mobility and uses a naming approach. By analysing these three protocols, we shed some light on how the success of each approach is very much dependent upon the mobile network scenario and its unique mobility conditions.

We have chosen to focus on route optimisation [8] as it is one of the key areas of concern with regards to NEMO and its real-life applications [2]. Several solutions to route optimisation for NEMO have been proposed, e.g. most recently [11], [13]. OptiNets builds upon NEMO and was designed to address its sub-optimal routing issue. We have chosen to compare with; (a) OptiNets, as there is prior analysis showing how it can improve the performance of NEMO; and (b) ILNPv6, which proposes a very different approach to mobility through changes in naming.

Our contribution is that by examining these three protocols, we present a comparative analysis of three different approaches to mobile networks, showing that there can be large differences in protocol overhead for the same scenario. Based on our results we propose that, although the NEMO approach is being standardised within the IETF, it may be beneficial for the research community to continue to examine other approaches.

We first provide a description of the tunnelling approach of NEMO and OptiNets (Section II), this is followed by a description of the naming approach adopted by ILNPv6 (Section III). We then compare NEMO, OptiNets and ILNPv6 by measuring the cost (overhead) of providing optimal routing for mobile networks; first using protocol-based analyses (Section IV-B) and second using an emulated passenger mobility traces (Section IV-C). This is followed by a summary of our work and a conclusion (Section V).

II. NEMO and OptiNets Overview

A. The NEMO protocol

NEMO enables network mobility by using an additional IP address, the *Care of Address (CoA)*, for the Mobile Router (MR). The CoA can be seen as a temporary address used by the MR as it moves. The CoA allows packets to be routed to the current location of the MR. The CoA acts as a locator. Meanwhile, the MR maintains another IP address that is available via DNS, its *Home Address (HoA)*, at its 'home network' (the IP sub-network to which the HoA belongs), and this is used for maintaining session state with Corespondent Nodes (CNs). The HoA acts as an identifier, and is used for transport layer state. When the MR is not at its home network, the Home Agent of the MR (HA_{MR}), acts as a proxy for the MR, forwarding packets received at the home network (using the HoA) to the MR (using the CoA), via a bi-directional, IP-in-IP tunnel. Traffic from within the mobile network is sent to

[1] https://datatracker.ietf.org/wg/mext/charter/

the MR. This traffic is encapsulated through this tunnel back to the HA where it is de-capsulated and forwarded. To CNs, the mobile network appears to be fixed.

This approach allows the MR and its Visiting Mobile Nodes (VMNs) to maintain pseudo-end-to-end connectivity despite changing network attachment points. A VMN achieves this by keeping its own Home Agent (HA_{VMN}) updated with its new CoA, using Mobile IPv6, as it moves. One benefit of this approach is that it does not change the way the IP address is used today. There are also no additional changes required of the IP architecture. The location of the mobile network is inconsequential so long as the MR and its HA_{MR} can set-up and maintain the bi-directional tunnel between them.

When a MR running NEMO migrates to a foreign network, it replies to any routing Advertisements it receives from the local *Access Gateway (AG)*, to receive a new CoA on the visited link. The MR then sends a Binding Update (BU) message to its HA_{MR}, informing it of its change of CoA (See Figure 1 step (1) and Figure 2(d)). The HA_{MR} updates its HoA-to-CoA cache for that MR and replies with a Binding Acknowledgement (BA). This act sets up and maintains the bi-directional tunnel between them.

Packets for the MR are received by the HA_{MR}, which uses IP-in-IP encapsulation to forward the packets to the MR at its latest CoA. All egress packets from the mobile network, sent from each VMN to its CN, must follow the same return path through the MR-HA_{MR} tunnel first before proceeding to its own respective HA_{VMN}(s) (See Figure 1 step (6)).

A mobile node has its own *Home Address* (HoA_{VMN}), which is always returned when a DNS lookup is performed for that mobile node. When this node becomes a VMN and joins a NEMO mobile network, it first receives its new CoA (Figure 1 step (2)). It then updates its HA_{VMN} with its CoA by sending a *Binding Update (BU)* message (See Figure 1 step (3) and Figure 2(a)). The HA_{VMN} responds with a *Binding Acknowledgement (BA)*. If the VMN is communicating with any MIPv6-aware CNs, it will execute a return routability test(RRT) (Figure 1 step (5a) and Figure 2(c)) and subsequently update its CNs with its new CoA, via a BU/BA exchange (Figure 1 step (5b) and Figure 2(b)).

Upon receiving its CoA, a VMN running MIPv6 maintains its own bi-directional tunnel between itself and its own HA_{VMN}. Operationally, the VMN-to-HA_{VMN} tunnel exists within the MR-to-HA_{MR} tunnel. Mobility of the MR and VMN is hidden as all traffic eventually is sent to/from their respective HA_{VMN}s.

If the MR changes location, it will again negotiate and receive its new CoA and update its HA_{MR} with its new location (Figure 1 step (4) and Figure 2(d)). The HA_{MR} then updates its Binding Cache and the bi-directional tunnel is maintained as it forwards MR packets to the new location.

As for the VMN within the mobile network, it will be unaware of its own mobility as the MR ensures that address on its ingress interface remain unchanged. The mobility of the MR only affects its egress interface. As a result, the VMN will not execute any handovers with its HA_{VMN} or its CNs(if any).

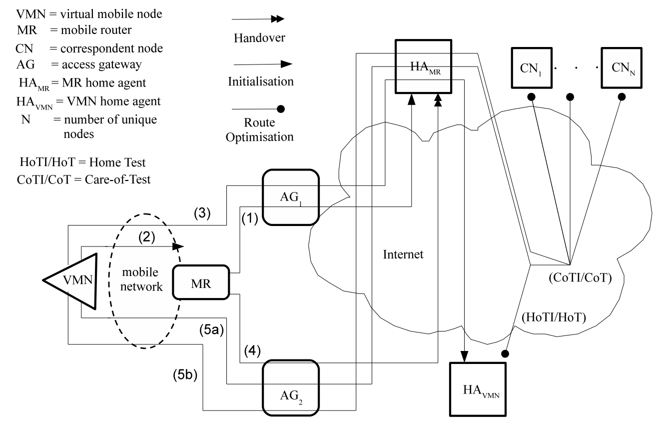


Fig. 1. The phases of initialisation and handover for a VMN (running Mobile IPv6) and MR (running NEMO). Step (1) shows the MR updating its HA_{MR} via AG_1. Step (2) shows a VMN arriving at the mobile network and registering an IP address gained from the MR. Step (3) shows the VMN updating its own HA_{VMN}. Step (4) shows the MR moving and conducting a handover by informing its HA_{VMN} of its new CoA. Step (5a) shows the VMN executing a RRT with its CNs. Step (5b) shows the VMN updating its CNs with a new CoA.

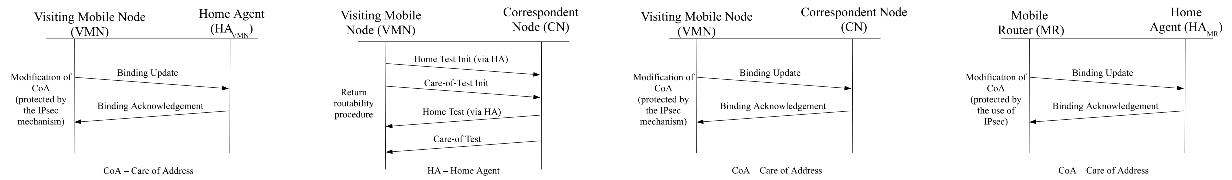
B. The OptiNets protocol

With the NEMO protocol, the VMNs within the mobile network are unaware of any mobility as the network prefix does not change. This topological inaccuracy (an unfortunate aspect of tunnelling) makes route optimisation by the VMNs running Mobile IPv6 impossible. The OptiNets protocol extends NEMO and makes this optimisation possible by having the MR advertise topologically correct network prefixes. As a result, all mobile nodes within the mobile network have topologically correct CoA(s). This allows VMNs running MIPv6 to execute Route Optimisation (RO) with RO-aware CNs, via a Return Routability Test (RRT) (Figure 4(b)) and a Binding Update (BU) (Figure 4(c)).

A mobile node has its own *Home Address* (HoA_{VMN}), which is always returned when a DNS lookup is performed for that mobile node. When this node becomes a VMN and joins a OptiNets mobile network, it must first receive its new CoA (Figure 3 step (2)). It then updates its HA_{VMN} with its CoA by sending a *Binding Update (BU)* message (See Figure 3 step (3) and Figure 4(a)). The HA_{VMN} responds with a *Binding Acknowledgement (BA)*. If the VMN is communicating with any MIPv6-aware CNs, it will execute a return routability test (RRT) (Figure 3 step (5a) and Figure 4(b)) and subsequently update its CNs with its new CoA, via a BU (Figure 3 step (5b) and Figure 4(c)). This is now possible as its address is topologically correct.

III. ILNPv6 Overview

The term *Identifier-Locator Network Protocol for IPv6 (ILNPv6)* is used, as it can be engineered as enhancements to IPv6 [1]. The operation of mobile networks within ILNPv6 is described in Figures 6 and 7. In ILNPv6, the end-system address is a dynamic binding between two parts: a topology-independent *Identifier (I)* value, and a topologically-significant



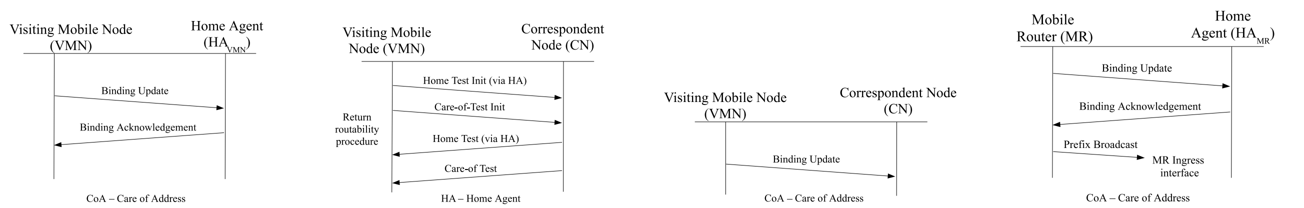
(a) VMN initialisation: 2 packets, total of 284 bytes

(b) Testing reachability: 4 packets, total of 536 bytes

(c) VMN-CN RO: 2 packets, total of 300 bytes

(d) MR handover: 2 packets, total of 228 bytes

Fig. 2. Timeline diagrams for a Visiting Mobile Node (VMN) and Mobile Router (MR) in NEMO. (a) corresponds to Figure 1 Step (3). (b) corresponds to Figure 1 Step (5a). (c) corresponds to Figure 1 Step (5b). (d) corresponds to Figure 1 Step (4).Packet sizes were obtained from [9].



(a) OptiNets VMN initialisation: 2 packets, 284 bytes

(b) OptiNets Testing reachability: 4 packets, 508 bytes

(c) OptiNets VMN-CN RO: 1 packet, 96 bytes

(d) OptiNets MR handover: 3 packets, 488 bytes

Fig. 4. Timeline diagrams for a Visiting Mobile Node (VMN) and Mobile Router (MR) in OptiNets extended NEMO. (a) corresponds to Figure 3 Step (3). (b) corresponds to Figure 3 Step (5a). (c) corresponds to Figure 3 Step (5b). (d) corresponds to Figure 3 Step (4). Packet sizes from [9].

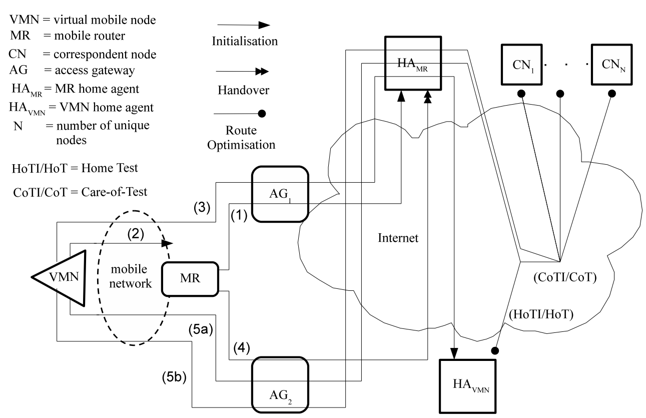


Fig. 3. The phases of initialisation and handover for a VMN (running Mobile IPv6) and MR (running OptiNets). Step (1) shows the MR updating its HA_{MR} via AG_1. Step (2) shows a VMN arriving at the mobile network and registering a topologically accurate IP address gained from the MR. Step (3) shows the VMN updating its own HA_{VMN}. Step (4) shows the MR moving and conducting a handover by informing its HA_{VMN} of its new CoA. This step also include the MR broadcasting its new Address Prefix to its Ingress interface. Step (5a) shows the VMN executing a RRT with its CNs. Step (5b) shows the VMN updating its CNs with a new CoA.

Locator (L) value. The Locator is not visible above the network layer and the upper layer state is bound only to the Identifier value. The end-system kernel maintains current I:L bindings for upper-layer sessions. The ILNPv6 Locator value uses the same semantics and bits as the the IPv6 address (routing) prefix (upper 64 bits), and so ILNPv6 packets pass transparently through the existing IPv6 core network. The ILNPv6 Identifier value occupies the lower 64 bits of the IPv6 interface ID, but has different semantics, identifying a *node* and not an *interface* (see Figure 5). The full 128-bits of the address (L:I) are used for Neighbour Discovery.

```
IPv6:
        | 3 |     45 bits         |  16 bits  |       64 bits              |
        +---+---------------------+-----------+----------------------------+
        |001|global routing prefix| subnet ID |    Interface Identifier    |
        +---+---------------------+-----------+----------------------------+

ILNPv6:
        |            64 bits                  |       64 bits              |
        +---+---------------------+-----------+----------------------------+
        |          Locator (L)                |   Node Identifier (I)      |
        +---+---------------------+-----------+----------------------------+
```

Fig. 5. IPv6 address (from RFC3587 [5]) as used in ILNPv6

A. Mobile networks with ILNPv6

ILNPv6 supports mobile networks natively [10]. In ILNPv6, the mobile network 'site' uses private addressing *internally* (to the site network) and the network's Mobile Router (MR) rewrites the Locator values of nodes within the site as packets transit that MR. This Locator re-writing does not affect end-system state (e.g. TCP connection state), as only the Identifier is used by the Transport layer. Nodes that are attached to the mobile network have DNS *LP* records that point to a common DNS *L64* record covering the entire mobile sub-network. The common *L64* record would be updated by the MR whenever its uplink moves to a different IPv6 network.

If we assume that our mobile network 'site' has an external link with Locator value L_1 (at access router AG_1), this will be held in a DNS *L64* record for the whole network. Each site will have a DNS *LP* which names points to the DNS *L64* record for the network. (Figure 6 step (1)). Within the mobile network, localised addressing is used through Locator rewriting in ILNPv6. A local (private) Locator value, L_L, is used by all nodes in the mobile network, and for all egress packets, the MR rewrites L_L to L_1, and performs the complementary operation for ingress packets. This is the ILNPv6 equivalent of NAT, but unlike IP, does not violate end-to-end state and is completely transparent to all ILNPv6 nodes. So, initialisation for a VMN occurs through a VMN receiving IPv6 Router Advertisements containing information about L_L and the *L64* record name for the mobile network, and the VMN updating its *LP* record to point to the *L64* record of the network (Figure 6 step (2) and Figure 7(a)).

When a handover is triggered for the link currently using L_1, a radio signal is detected in the new cell and a new Locator value, L_2, is obtained from the Access Gateway (AG_2). This can be done through normal IPv6 discovery mechanisms, as Locator values are identical to IPv6 network prefixes. We will assume that the radio cells providing L_1 and L_2 overlap. Then, the MR updates the DNS *L64* record (currently holding value L_1) to value L_2 (for new sessions) (Figure 6 step (3). Figure 7(b)). It then starts updating the state of existing sessions using value L_1, to using the new value of L_2, by issuing *Locator Update (LU)* messages (synonymous to Binding Update message in IPv6) for CNs using L_1 (Figure 6 step (4) and Figure 7(c)). It then transitions sessions from L_1 to L_2 using Locator rewriting. When no more packets arrive from remote locations using L_1 within a given time period (i.e. all sessions have made the transition to L_2), the connection is considered to have completed handover. This is a *soft handover* at the ILNPv6 layer, something that is not currently defined for MIPv6 or NEMO. The MR is providing this capability efficiently for the whole mobile network. Note also that during this time, it would also be possible to have another MR and have the whole mobile network multi-homed [10].

It is also possible to use ILNPv6 for normal handover, simply by switching to L_2 as soon as possible. Any packets in flight addressed to L_1 may be lost, but can be recovered through the retransmission capability in TCP, for example, albeit this would be inefficient, as it may invoke the congestion control behaviour of TCP through lost/delayed TCP data segments or ACKs.

IV. Emulation

We formulated general scenario independent equations for packet and bandwidth cost, as presented in Equations (4) to (12). We then used statistical data of train movements and emulated data of passenger movements to compare the performance cost of NEMO, OptiNets and ILNPv6 in a mobile network scenario. Our scenario is that of the London Circle Line, which is a line on the public metropolitan rail system in the heart of London, UK. We have assumed that passengers boarding and leaving the Circle Line trains are VMNs. We considered each train as a separate mobile network, and each arrival of the train at a new station as a movement of the mobile network that requires it to establish a new network point of attachment. Note that this is *not* a simulation study. We have not used a mobility model and do not maintain state for individual nodes in our evaluation. We use the Circle

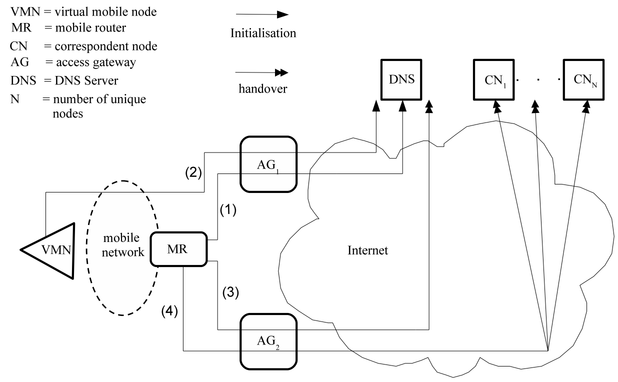


Fig. 6. This figure shows the 2 phases of Initialisation and Handover for a VMN and MR for ILNPv6. Step (1) shows the MR arriving at a new location, receiving an address from AG_1 and updating its DNS L record with its latest location. Step (2) shows a VMN arriving at the mobile network, receiving a new (local) Locator and name of the L record for the network, then updating its DNS LP record. Step (3) shows the MR moving to a new location, receiving a new Locator from AG_2, and updating its DNS L record with this new location. Step (4) shows the MR updating all existing sessions between its VMNs and their CNs.

Hours of service per day (N_d)	18
No. of trains per station per hour (N_t)	7
No. of stations per hour (N_s)	27
Mean no. of passengers on a weekday (N_w)	218136

TABLE I
London Circle Line Data from Tubeprune [12].

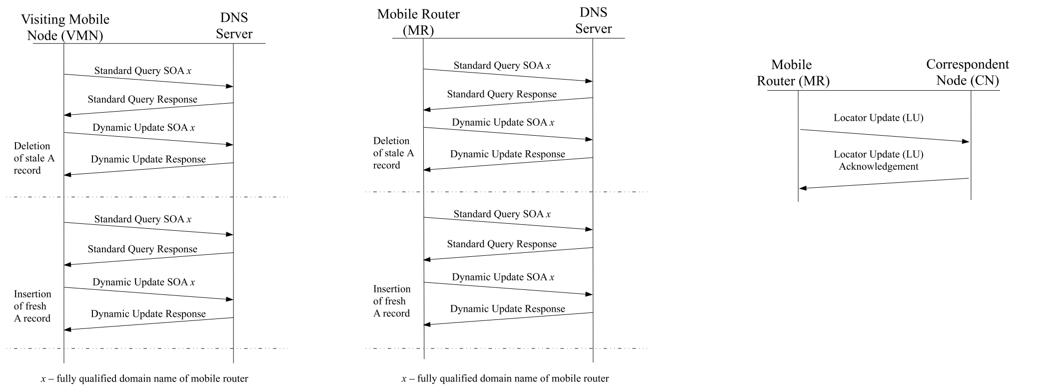
Line data to provide realistic input numbers, rather than chose arbitrary values.

The purpose behind our experiments is to bring to light the cost of providing for route optimisation (for mobile networks) with different approaches. We chose to factor out the contribution of the wireless layer in our experiments. As we are interested only in the architectural differences between the Naming and Tunnelling approaches, exclusion of any wireless effects allows us to confidently draw conclusions based on differences in protocol architecture only. This provides a constrained *comparison* (based on fewer variables). This also makes our emulation less complex. However, this is not an *absolute* performance analysis: simulations for specific scenarios (e.g. use of WLAN, WiMAX or 3G for the MAC/PHY) would be required for *operational* evaluations.

A. Data for the emulation

Two sets of data were required, first was the statistical data regarding the London Circle Line. This was collected from Tubeprune [12] and Transport for London [4]. The second was the protocol exchanges for the NEMO/MIPv6 updates of HAs. These were obtained from the previous detailed analysis in [9]. These resulted in the timeline diagrams in Figure 4. The raw data we used from Tubeprune is summarised in Table I. Our derived data is summarised in Table II.

We began with an equal number of passengers onboard each train and at each train station. $N_p = (N_w/N_d)/(N_t+N_s) = 356$.



(a) VMN initialisation: 8 packets, 1362 bytes

(b) MR Handover: 8 packets, 1362 bytes

(c) Update of CN: 2 packets, 144 bytes

Fig. 7. Timeline diagrams for a Visiting Mobile Node (VMN) and Mobile Router (MR) in ILNPv6. (a) corresponds to Figure 6 Step (2); (b) corresponds to Figure 6 Step (3); (c) corresponds to Figure 6 Step (4). Packet sizes were obtained from inspecting DNS message exchanges using *tcpdump*.

No. of passengers (VMNs) per hour per train (N_p)	356
Handover/stop time at stations per train (T_h)	60s

TABLE II
PASSENGER AND TRAIN MOVEMENT USED IN EMULATION.

So, at the start of our emulation, there are 356 passengers in each train, and 356 passengers at each train station waiting to board. We assume that this number stays constant throughout the experiment, i.e. passengers only make a single journey during the day, so VMN initialisation only happens once per passenger per train per day.

The variables we consider in this experiment are (i) the number of stations a passenger travels through (i.e. handovers, N_h) and (ii) the number of unique CNs per train (N_{CN}). For OptiNets, (i) affects the VMN handovers (Figure 4(a)), and (ii) affects the number of VMN-to-CN updates (Figure 4(b)). There are also a number of packets which have to be generated due to VMN initialisations and MR-to-HA handovers, regardless of (i) or (ii). VMN initialisations are dependent on the number of passengers (N_p). MR-to-HA handovers are dependent on the number of train stations (N_s). We assumed that handovers and initialisations of VMNs occur during the time that the train is at a station (T_h).

Looking at Equations (4) to (12), we see that they share the same denominator ($N_p.N_s.T_h$). This term is defined as the timeframe in which all registrations and handovers must be completed per passenger per station, i.e. the period that two cells would overlap is the hand-off period. We have defined T_h as 60s (the average time a train spends at each station) and assumed that handovers occur at the station. The output of each equation will be framed as the the cost in terms of Packet overhead and Bandwidth Overhead, separately, per passenger per station, evaluated during the handover period, T_h. We have also assumed that the NEMO protocol has enabled IP Authentication Header [6].

B. General Analysis

1) NEMO Analysis: The overhead generated by NEMO per passenger per train per second, C_{NEMO}, is calculated as:

$$C_{NEMO} = \frac{K_1.N_p + K_2.N_p.N_{CN} + K_3.N_s}{N_p.N_s.T_h} \quad (1)$$

where K_1, K_2 and K_3 are constants. There are three parts to the right-hand side of the numerator of this expression. The first part ($K_1.N_p$) refers to the VMN initialisation (see Figure 2(a)). As each VMN will have to join the mobile network, and we have assumed each passenger only makes one trip, thereby joining only once, this number is dependent only on the number of passengers (N_p).

In the second part, we assume VMN executes route optimisation with its existing CNs. This include a return routability test (RRT) (See Figure 1 step (5a) and Figure 2(c)) and a BU/BA (See Figure 1 step (5b) and Figure 2(b)).

In the third part of the expression, ($K_3.N_s$) is the overhead generated by the MR (see Figure 2(d)) per train per hour for a handover. Given that there are 27 stations, N_s, on this route, and that each train takes approximately an hour (on average) to finish one circuit we calculate the number of MR handovers generated per train per hour to be $K_3.27$.

In Eqn 10, if we replace K_1, K_2 and K_3 with the appropriate packets counts or byte counts, we get, respectively, the packet overhead, N_{NEMO}, and bandwidth overhead, B_{NEMO}:

$$N_{NEMO} = \frac{2.N_p + 6.N_p.N_{CN} + 2.N_s}{N_p.N_s.T_h} \quad (2)$$

$$B_{NEMO} = \frac{284.N_p + (536+300).N_p.N_{CN} + 228.N_s}{N_p.N_s.T_h} \quad (3)$$

2) OptiNets Analysis: The overhead generated by OptiNets per passenger per train per second, C_{OPTI}, is calculated as:

$$C_{OPTI} = \frac{H_1.N_p + H_2.N_p.N_{CN}.N_h + H_3.N_s}{N_p.N_s.T_h} \quad (4)$$

where H_1, H_2 and H_3 are constants. There are three parts to the numerator of the right-hand side of this expression. The first part ($H_1.N_p$) refers to the VMN initialisation (Figure 4(a)). As each VMN will have to join the mobile network, and we have assumed each passenger only makes one trip, thereby joining only once, this number is dependent upon the number of passengers (N_p). For the second part, we assume VMN executes route optimisation with its existing CNs. This includes a return routability test (RRT) (Figure 3 Step (5a) and Figure 4(b)) and a BU (Figure 3 Step (5b) and Figure 4(c)), every time the MR changes location and thus cannot exceed the total number of train stations (N_s). For the third part, we evaluate the overhead generated by the MRs (Figure 4(d)). Given that there are 27 stations (N_s), on this route, and that each train takes approximately an hour (on average) to finish one circuit, we calculate the number of MR handovers generated per train per hour to be $H_3.27$.

In Eqn 4, we replace H_1, H_2 and H_3 with the appropriate packets counts or byte counts from Figure 4, we get, the packet overhead, N_{OPTI}, and bandwidth overhead, B_{OPTI}:

$$N_{OPTI} = \frac{2.N_p + 5.N_p.N_{CN}.N_h + 3.N_s}{N_p.N_s.T_h} \quad (5)$$

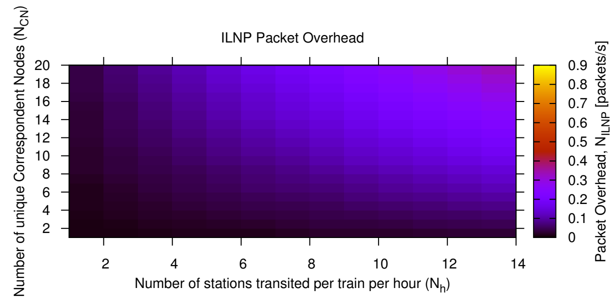
$$B_{OPTI} = \frac{284.N_p + (508 + 96).N_p.N_{CN}.N_h + 488.N_s}{N_p.N_s.T_h} \quad (6)$$

3) ILNPv6 Analysis: The overhead generated at handover by ILNPv6 per passenger per train per second, C_{ILNP}, is calculated as:

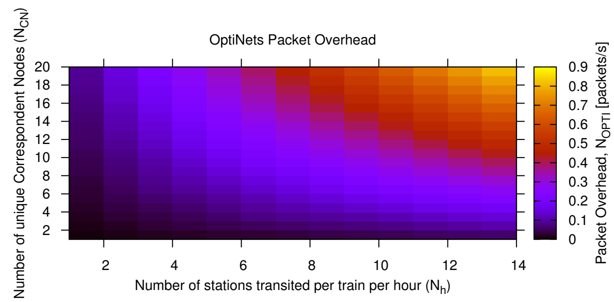
$$C_{ILNP} = \frac{J_1.N_p + J_2.N_s + J_3.N_p.N_{CN}.N_h}{N_p.N_s.T_h} \quad (7)$$

where J_1, J_2, and J_3 are constants. There are three parts to the numerator of the right-hand side of this expression. The first part ($J_1.N_p$) refers to the VMN initialisation (Figure 7(a)). As each VMN will have to join the mobile network, and we have assumed each passenger only makes one trip, thereby joining only once, this number is dependent only on the number of passengers (N_p). The second part is the overhead generated by the handover of the MR, updating its location (Figure 7(b)). This is dependent only upon the number of train stations visited along the route (N_s). The third part is the overhead generated by the MR to each unique CN of the resident VMNs to update existing sessions (Figure 7(c)). This is directly dependent on the number of passengers (N_p), the total number of unique CNs for the mobile network (N_{CN}) as well as the number of train station handovers (N_h).

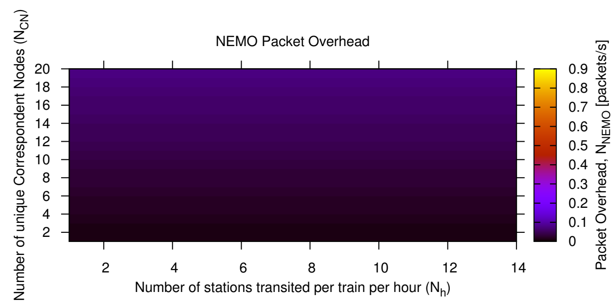
In Eqn 7, we replace J_1, J_2 and J_3 with the appropriate packets counts or byte counts from Figure 7, we get the packet overhead, N_{ILNP}, and bandwidth overhead, B_{ILNP}:



(a) ILNPv6 - Eqn (8) (min/max: 0.01/0.35)



(b) OPTI - Eqn (5) (min/max: 0.01/0.87)



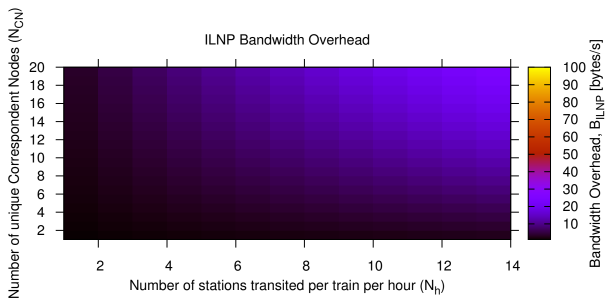
(c) NEMO - Eqn (11) (min/max: 0.01/0.08)

Fig. 8. These are derived from Eqns 8, 5, 11. The horizaontal axis is the no. of stations transited per train per hour and the vertical axis is the packet overhead packet overhead per person [packets/s]. A darker shade represents less packet overhead.

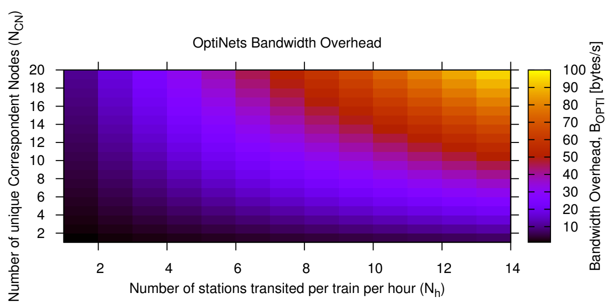
$$N_{ILNP} = \frac{8.N_p + 8.N_s + 2.N_p.N_{CN}.N_h}{N_p.N_s.T_h} \quad (8)$$

$$B_{ILNP} = \frac{1362.N_p + 1362.N_s + 144.N_p.N_{CN}.N_h}{N_p.N_s.T_h} \quad (9)$$

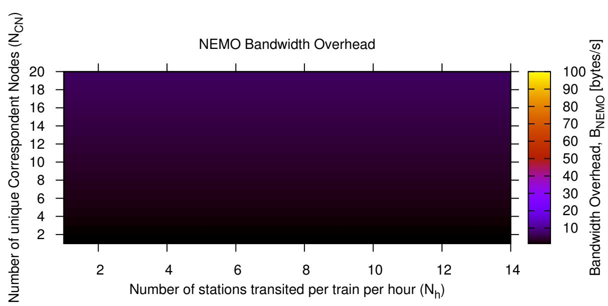
4) Results: We have already noted above some key architectural differences when comparing the NEMO and ILNPv6 approach. From Figures 3 and 6, we see that the protocol exchange for ILNPv6 is simpler than for NEMO/OptiNets, and the data path that results is also simpler, compared with NEMO which requires two sets of tunnels. Additionally, we find that ILNPv6 leverages existing DNS infrastructure for naming, whilst NEMO/OptiNets must introduce additional network entities (the HA and FA) in order to function. Also, the use of the tunnels creates potential inefficiency in packet forwarding, and system complexity, as a result of tunnels and



(a) ILNPv6 - Eqn (9) (min/max: 0.99/25.79)



(b) OPTI - Eqn (6) (min/max: 0.57/104.59)



(c) NEMO - Eqn (12) (min/max: 0.70/10.51)

Fig. 9. These are derived from Eqns 9, 6,12. The horizaontal axis is the no. of stations transited per train per hour and the vertical axis is the packet overhead bandwidth overhead per person [bytes/s]. A darker shade represents less bandwidth overhead.

redirection through home networks.

Using our expressions for packet overhead in Eqn (5), and our expressions for bandwidth overhead in Eqn (6), we vary the value of N_h from 1 to 14 (half a circuit of the Circle Line), and the value of N_{CN} from 1 to 20 (i.e. every VMN has 20 *unique* CNs). For reference, we also include the overhead for NEMO/OptiNets without route-optimisation, i.e. all VMNs using tunnels via the MR_{HAR}:

$$C_{NEMO} = \frac{K_1.N_p + K_2.N_p.N_{CN} + K_3.N_s}{N_p.N_s.T_h} \quad (10)$$

$$N_{NEMO} = \frac{2.N_p + 2.N_p.N_{CN} + 2.N_s}{N_p.N_s.T_h} \quad (11)$$

$$B_{NEMO} = \frac{284.N_p + (300).N_p.N_{CN} + 228.N_s}{N_p.N_s.T_h} \quad (12)$$

From Figure 8, we see that, compared to NEMO, the packet overhead of OptiNets is much greater - a factor of $\sim$10. From Figure 9, we see similar increases for bandwidth overhead - a factor of $\sim$10 for OptiNets compared to NEMO.

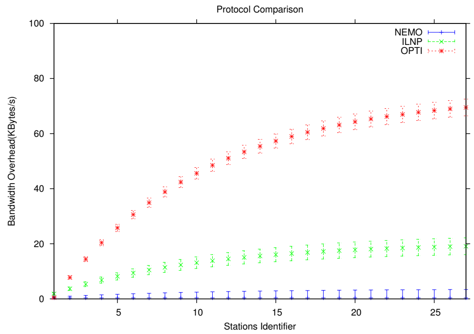
C. Emulation with passenger movement

Looking at the general equations from the first emulation, we see that the common unknown is the duration that each passenger stays on the railway system, i.e. the Number of station hops (N_h). This is because of the lack of individual passenger mobility traces. The train statistics do not have this level of granularity. So, we have combined the available statistics, along with the assumption of a uniform random distribution for passengers (with each passenger having 2 CNs) to obtain values of the number of handovers for each passenger. Even though this is not a real mobility model, as our study overall is *comparative*, and the same model is applied to OptiNets and NEMO, we believe that it is sufficient.

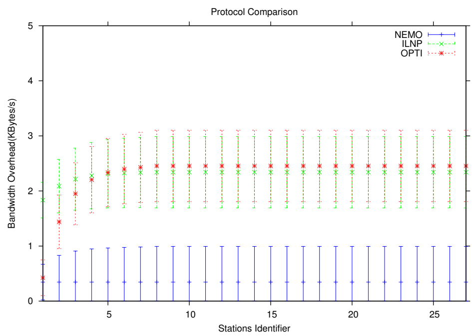
Our emulation (written in Java) models the London Circle line using Equations (4) to (12). Our software emulates all 27 train stations as well as the individual train arrivals and departures. It also models each specific passenger boarding the train at a random selected train station, and alighting at a randomly selected stations. Specifically, the train movements and passenger numbers follow the available statistics. For the passenger movements, we used the total number of passengers for a year and divided this equally to all train stations for a year (taking into account the passenger differences of weekdays, saturdays and sundays). We then used a uniform random distribution to emulate the arrival of passengers from this pool for a given day and train station. As a train arrives, the passengers board that train and at each subsequent stop, we randomly select passengers to alight the train. This selection is based on the passenger mobility ratio, R_P, which we define as the ratio of number of passengers in the train that alight to the number that remain on board. For the purposes of our emulation, we used three different values of R_P: 10%, 50% and 90%. In total, we executed each emulation 3 times, each with a different value of R_P. Each run emulates one year of train and passenger movements. We then used the corresponding results to revisit our original general equations of bandwidth for NEMO, OptiNets and ILNPv6. The results are shown in Figures 10(a), 10(b) and 10(c) respectively.

One outcome of assuming a uniform random passenger distribution is that the average number of passengers registering to the mobile network is constant through each run. We note that the distribution of passenger arrival has no noticeable effect on N_h: N_h is directly dependent upon the passenger movement ratio, which is consistent with a uniform random distribution. Higher values for R_P (more passengers leave/join the network), result in fewer average number of handovers per train station (lower values for N_h).

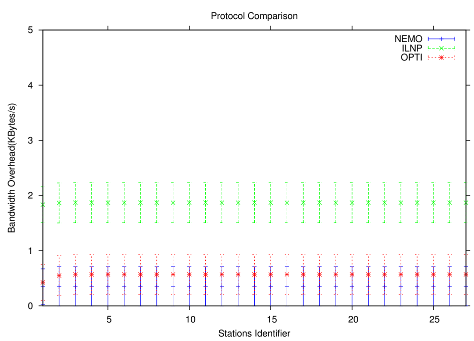
We also note that depending on the rate of passengers boarding and alighting the train, there is a steady state where the number of handovers per stop becomes stable. The lower the difference between these two rates, the quicker the steady state is achieved.



(a) 10% Passenger Movement Ratio



(b) 50% Passenger Movement Ratio



(c) 90% Passenger Movement Ratio

Fig. 10. Bandwidth overhead per train [Kb/s], when CN=2

By observing how OptiNets and ILNPv6 behave with different values of R_P, we see that ILNPv6 is much less sensitive to the change in number of handovers (N_h), compared to OptiNets. When R_P is set at 50% (Figure 10(b))), both ILNPv6 and OptiNets have similar overhead costs. However, when R_P is set at 90% (Figure 10(c)), the cost of OptiNets increases by an order of magnitude in comparison to the increase of ILNPv6. As a result, in mobile scenarios of high passenger mobility fluctuations, the OptiNets approach will possibly lead to much larger variations in bandwidth usage compared to ILNPv6.

V. Conclusion

We have compared NEMO, OptiNets and ILNPv6, which are three different approaches to optimised routing for mobile networks. We have created an emulation of a network scenario, focussing on *initialisation* and *handover*, and derived analytical expressions for packet overhead and bandwidth overhead.

By using a scenario based on data from passengers and train numbers on the Circle Line metropolitan railway in London, UK, we have evaluated the expressions for packet overhead and bandwidth overhead by varying two key characteristics of a mobile network; its degree of mobility, N_h (which we measure by the number of handovers as trains move between stations), and the number of external communications from the mobile network, N_{CN} (which we emulate by the number of unique CNs for each VMN).

We have quantified the cost of providing for routing in terms of packet and bandwidth overhead by deriving analytical expressions for NEMO, OptiNets and ILNPv6. With these expressions, we chose two variables N_h and N_{CN} to test their respective performance. We then wrote a Java-based emulation to generate the effects of passenger mobility with the assumption of a uniform random distribution of passengers arrivals (and assuming N_{CN} is 2) to calculate an approximate value of N_h.

We have shown that, with respect to control overhead, there exists a trade-off between provisioning for mobility and having optimal routing paths for mobile traffic flows. NEMO may be better suited for mobile networks that do not have mobility aware nodes. For OptiNets, we see that RO for one level of tunnelling has a higher overhead in comparison and would be worse if multiple levels of tunnelling exist. ILNPv6, may be better suited where the mobile network is a mix of mobile and static nodes.

References

[1] R. Atkinson, S. Bhatti, and S. Hailes. A Proposal for Unifying Mobility with Multi-Homing, NAT, and Security. In *ACM MobiWAC2007*, Greece, Oct. 2007.

[2] R. Baldessari, T. Ernst, A. Festag, and M. Lenardi. Automotive Industry Requirements for NEMO Route Optimization. Internet Draft.

[3] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert. Network Mobility (NEMO) Basic Support Protocol. RFC 3963 (Proposed Standard), Jan. 2005.

[4] T. for London. Circle line facts. www.tfl.gov.uk/tfl/corporate/modesoftransport/tube/linefacts/?line=circle.

[5] R. Hinden, S. Deering, and E. Nordmark. IPv6 Global Unicast Address Format. RFC 3587, August 2003.

[6] S. Kent. IP Authentication Header. RFC 4302, Dec. 2005.

[7] E. Perera, A. Seneviratne, and V. Sivaraman. Optinets: An architecture to enable optimal routing for network mobility. In *International Workshop on Wireless Ad-hoc Networks*, 2004.

[8] E. Perera, V. Sivaraman, and A. Seneviratne. Survey on Network Mobility Support. *SIGMOBILE Mobile Computer Communication Review*, 8(2):7–19, 2004.

[9] H. Petander, E. Perera, K.-C. Lan, and A. Seneviratne. Measuring and Improving the Performance of Network Mobility Management in IPv6 Networks. *IEEE J. on Selected Areas in Communications*, 24, 2006.

[10] D. Rehunathan, R. Atkinson, and S. Bhatti. Enabling Mobile Networks Through Secure Naming. In *Proc. MILCOM2009 - 28th IEEE Military Communications Conference*, 2009.

[11] I. Soto, C. Bernardos, M. Calderon, A. Banchs, and A. Azcorra. NEMO-enabled localized mobility support for Internet access in automotive scenarios. *IEEE Communications*, 47(5):152–159, May 2009.

[12] Tubeprune. London underground statistics. www.trainweb.org/tubeprune/Statistics.htm.

[13] F. Z. Yousaf, A. Tigyo, and C. Wietfeld. Neron: A route optimization scheme for nested mobile networks. In *WCNC2009*, Apr 2009.