Optimal Distribution of Remotely-Subscribed Multicast Traffic within a Proxy Mobile IPv6 Domain by Using Explicit Multicast

Luis M. Contreras *, Carlos J. Bernardos †
* Core Network Evolution
Telefónica I+D
Don Ramón de la Cruz, 82-84, 28006, Madrid, Spain
† Departament of Telematics Engineering,
Universidad Carlos III de Madrid
Avda. Universidad, 30, 28911, Leganés, Spain.
lmcm@tid.es, cjbc@it.uc3m.es

Abstract— **Distribution of remotely subscribed multicast content in a Proxy Mobile IPv6 (PMIPv6) domain is performed by means of bi-directional IP-in-IP tunnels established between the mobility anchor and the visited access gateways where the mobile terminals consuming such traffic are attached to. Each access gateway subscribing to content on behalf of an attached mobile terminal requires a separate copy of the remote multicast flow being distributed over the PMIPv6 domain. In many cases, these individual copies traverse the same routers in the path from the mobility anchor towards the access gateways, incurring in an inefficient distribution, which is equivalent to the unicast delivery of the remote multicast content within the domain. This paper explores the potential gain obtained by using explicit multicast instead of the standard IP-in-IP tunneling, showing relevant capacity savings with lower overhead respect to the standard distribution case. This transport service based on explicit multicast emerges then as an attractive transport alternative for PMIPv6 domain operators serving visiting mobile multicast consumers.**

Keywords-component; PMIPv6; multicast; xcast; optimization.

I. INTRODUCTION

The new capabilities being offered by the mobile wireless technologies are bringing broadband capacity networks outside the home, representing additional delivery options for the distribution of broadband services on the move. The commercial success of mobile multimedia-enabled terminals, mainly because of the success of iOS and Android based devices, is rapidly increasing the demand of mobile data access, especially audiovisual contents.

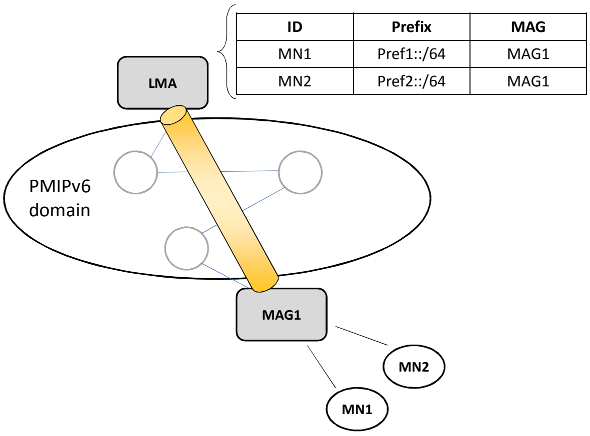
IP multicast basically facilitates the delivery of a single copy of a data stream to multiple listeners interested in receiving the same content simultaneously. This capability is commonly used nowadays in telecom networks, for instance, to distribute TV content (known as IPTV service). The need for supporting the same kind of services both in fixed and mobile networks brings the necessity of delivering IP multicast also to mobile receivers.

Proxy Mobile IPv6 (PMIPv6) [1] is a network-based mobility management protocol which enables the network to provide mobility support to standard IP terminals residing in the network. These terminals enjoy this mobility service without being required to implement any mobility-specific IP operations. Namely, PMIPv6 is one of the mechanisms adopted by the 3GPP to support the mobility management in future Evolved Packet System (EPS) networks [2].

PMIPv6 allows a Mobile Access Gateway (MAG) to establish a distinct bi-directional tunnel with different Local Mobility Anchors (LMAs), being each tunnel shared by the attached Mobile Nodes (MNs). Each mobile node is associated with an LMA, which keeps track of its current location, that is, the MAG where the mobile node is attached. IP-in-IP encapsulation is used within the tunnel to forward traffic between the LMA and the MAG (see Figure 1).

The basic solution [3] of multicast traffic distribution within a PMIPv6 domain makes use of the bi-directional LMA-MAG tunnels. It follows the so-called remote subscription model, in which the subscribed multicast content is delivered from the Home Network. By doing so, an individual copy of every multicast flow is delivered through each tunnel connecting the mobility anchor to any of the access gateways in the domain. In many cases, these individual copies traverse the same routers in the path towards the access gateways, incurring in an inefficient distribution, equivalent to the unicast distribution of the multicast content in the domain, as shown in Figure 2.

This fact leads to distribution inefficiencies and higher per-bit delivery costs, incurred by a PMIPv6 domain operator offering transport capabilities to a Home Network operator for serving their MNs when attached to the PMIPv6 domain. As long as the remotely subscribed multicast service is not affected, it seems worthy to explore more optimal ways of distributing such content within the PIMPv6 domain.



ID	Prefix	MAG
MN1	Pref1::/64	MAG1
MN2	Pref2::/64	MAG1

Figure 1. PMIPv6 mode of operation

As later discussed, local multicast distribution to the PMIPv6 domain (also known as *direct routing*) is not always feasible then we focused on the remote subscription case. This paper addresses this issue by analyzing the capabilities provided by Explicit Multicast (Xcast) [4] to provide an optimal and efficient multicast traffic distribution from the bandwidth consumption point of view. Section II describes the different alternatives existing today for multicast traffic distribution within a PMIPv6 domain, remarking the potential inefficiency observed in case of remotely-subscribed multicast. Section III introduces the applicability of the explicit multicast for the distribution of the multicast flows in the domain, and extensions needed for using explicit multicast in PMIPv6 are identified. Furthermore, in Section IV a performance evaluation is conducted to assess the potential gains due to the use of explicit multicast in the distribution network. Section V addresses some conclusions and advances some further work. Finally, Appendix A provides some insights on the PMIPv6 domain scalability to determine the viability of the proposed explicit multicast approach.

II. MULTICAST DISTRIBUTION IN PMIPV6

As a general procedure for subscribing to a multicast content, a mobile node expresses its interest in joining or leaving a multicast group by sending Multicast Listener Discovery (MLD) control messages to the MAG, which acts as the first hop at the point-to-point link established with the MN. The MAG maintains the individual multicast status of the interface for that link and handles the multicast traffic towards the MN accordingly to the MLD messages received. There are two alternatives to distribute multicast traffic within a PMIPv6 domain: remote subscription and direct routing.

The former is primary focused on the multicast distribution from networks outside the PMIPv6 domain (e.g., the Home network or third parties networks), while the later results convenient for the multicast distribution of content locally available at the PMIPv6 domain.

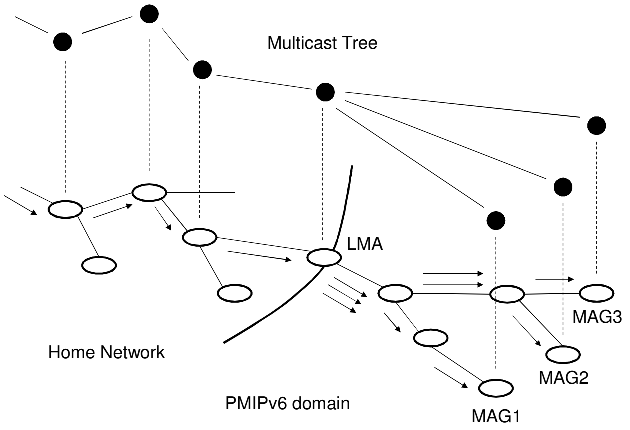


Figure 2. Inefficient distribution of remote subscribed multicast traffic in a PMIPv6 domain

A. *Remote subscription*

The baseline solution [3] considers only the remote subscription case, where the MN obtains the desired multicast stream from its home network, through the local mobility anchor. The LMA is in charge of interacting with the multicast infrastructure out of the PMIPv6 domain.

In the base solution, the MAG instantiates a distinct MLD proxy functionality per every set of MNs associated to a specific LMA. Each MLD proxy instance is responsible of summarizing the subscription requests of the MNs connected to it on a per LMA association basis. The different proxy instances of the same MAG are isolated one from the other.

With the remote subscription model, the multicast traffic reaches the MNs after going through the corresponding LMA (note that there might be multiple LMAs in the same domain). For every proxy instance in the MAG, the tunnel interface pointing to the LMA becomes the proxy upstream interface, whereas the links towards the MNs are the corresponding downstream interfaces of each instance.

Then, every MAG-LMA tunnel is part of a separate MLD proxy domain, being a branch of the multicast tree built for multicast traffic distribution internally to the domain. A single copy of a data stream will be sent per group of MNs (attached to a certain MAG) associated to the same LMA. The LMA will maintain the multicast state of every tunnel interface, reflecting the summarized view offered by the MAG on behalf of the attached MNs bound to the LMA.

The base solution suffers from the tunnel convergence problem, where several copies from the same multicast stream can reach the access gateway when simultaneous subscriptions from MNs associated to distinct LMAs occur.

To avoid that, a central entity named Multicast Tree Mobility Anchor (MTMA) [5] can be deployed in the PMIPv6 domain to act as the topological anchor point for remotely serving multicast traffic to the MNs in the domain, independently of the LMA which maintain the association for receiving unicast traffic.

The MTMA connects to the MAG as described in [3]. The bi-directional tunnels among the MTMA and the access gateways in the domain are part of the multicast tree for remote multicast traffic distribution. Therefore, a copy of every multicast channel subscribed by a MAG on behalf of an attached MN is transported on those tunnels to reach the corresponding access gateway. The MTMA can be then considered as a form of upstream multicast router with tunnel interfaces allowing remote subscription for the MNs.

B. Direct routing

A second option to limit the number of copies of the same content at the MAG is the usage of a native multicast infrastructure in the PMIPv6 domain [5] allowing direct multicast routing from locally available multicast sources. In this case, the MAG can be directly connected to an upstream multicast router in the PMIPv6 domain, while the unicast traffic remains served as normally by the corresponding LMAs.

Following this approach, the usage of the bi-directional tunnels is totally avoided, since the multicast traffic is natively distributed within the PMIPv6 domain. This is the most effective way of multicast distribution within the domain, but unfortunately it is not always possible for non-technical reasons, such as for example:

- The multicast source is not local to the PMIPv6 domain, being located either in the Home network or hosted by a third party.
- The multicast content cannot be natively distributed within the local PMIPv6 domain due to administrative or regulatory reasons; as for instance, multicast address allocation issues between the assigned addresses in the local PMIPv6 domain and in the multicast source home network (i.e., a certain multicast IP address identifies different multicast content channels in both the Home and the PMIPv6 domains), or some contents may be not allowed for distribution in a certain network, like regional or ethnical channels out of the target region.
- The multicast content is not natively distributed in the local PMIPv6 domain due to commercial and business intelligence reasons; for instance, the Home network operator might not be interested on providing visibility about what content its MNs subscribe to.

These are some of the reasons why the remote subscription case is relevant and requires to be properly addressed. PMIPv6 domain operators can commercialize this service, offering transport capabilities to the Home network operators to reach its MNs with a multicast service. Providing this transport service in the most efficient manner is then economically attractive from the PMIPv6 domain operator point of view.

C. Efficiency problems

The transport of the remotely-subscribed multicast traffic by means of IP-in-IP unicast tunnels in the PMIPv6 domain is inefficient as several copies of the same content traverse the same links and are forwarded by the same routers. Two alternatives to improve this distribution can be taken into account: native multicast transport (direct routing) on the PMIPv6 domain, or explicit multicast (Xcast) transport of the multicast traffic. The former has been already described, and some situations could prevent its use. We now focus on the latter, by proposing the use of IP-in-Xcast encapsulation between the mobility anchor and the access gateways instead of the standard IP-in-IP tunneling.

III. MULTICAST DISTRIBUTION AMONG MOBILITY ANCHOR AND ACCESS GATEWAYS WITH EXPLICIT MULTICAST

A. Introduction to Xcast

The Xcast protocol has been proposed as a way of optimizing the delivery of multicast traffic for small groups. Basically, the Xcast mechanism eliminates the need of per-session signaling and per-session state information of traditional IP multicast schemes by including the list of destinations in the data packet, instead of using a multicast address. To do that, the source node keeps track of the final destinations in the multicast channel that it wants to send packets to.

With Xcast, each router in the path between the source and the destination parses the header and creates a new datagram for every next hop including only the destinations reachable through that next hop according to the routing table, in such a way that the header of the subsequent Xcast packets only contains the destinations available in the path. The Xcast packet always follows the ordinary unicast routing for a given destination.

When just one destination remains to be reached, the Xcast packet is transformed into a normal unicast packet. Figure 3 graphically describes the Xcast procedures for the case in which a node A simultaneously delivers data content to nodes B, C and D with Xcast encapsulation.

The processing that a router does for every Xcast packet is the following: *(i)* the router performs a route table lookup to determine the next hop for each of the destinations listed in the packet; *(ii)* the router partitions the set of destinations based on their next hops; *(iii)* it replicates the packet so that there is one copy of the packet for each of the next hops found in the previous steps; *(iv)* before delivering the new packet, it modifies the list of destinations in each of the copies so that the list in the copy for a given next hop includes just the destinations reachable through that next hop; *(v)* finally, the router sends the modified copies of the packet on the next hops.

B. Benefits and impacts of using Xcast

Regarding traditional multicast, Xcast offers a number of advantages that have been reported on [4], such as not needing to maintain multicast state per group in every router on the tree, or not requiring multicast address allocation. However, some drawbacks have also been identified, such as the incurred overhead, or the header processing complexity. Furthermore, as described later in the paper, the specified Xcast header allows a maximum of 127 destinations. This means that in case of having more destinations on the path, separate Xcast trees should be formed.

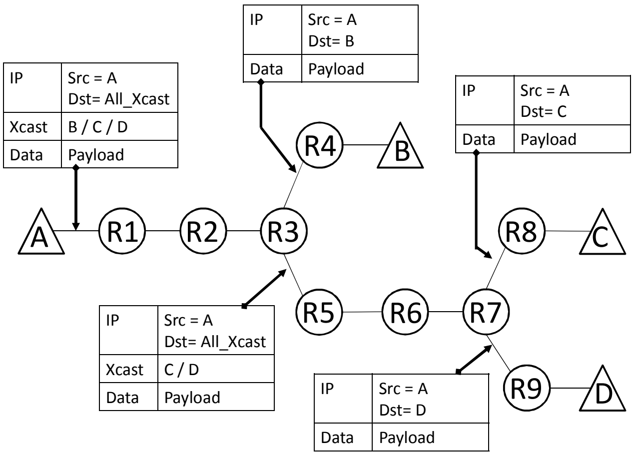


Figure 3. Explicit multicast (Xcast) mode of operation

This implies that, when applied to the PMIPv6 case, a limit of 127 access gateways per mobility anchor's downstream interface has to be considered in the distribution tree. In the Appendix A the number of access gateways in a domain is discussed in order to determine how this limit could impose restrictions in a typical PMIPv6 domain for the deployment of Xcast functionalities.

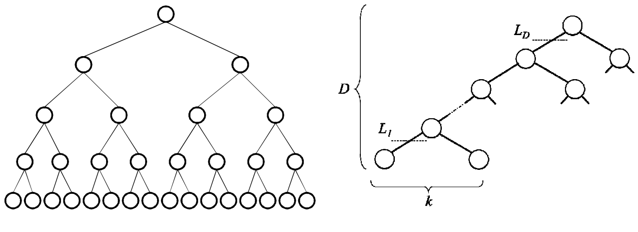
C. *Modifications to standard PMIPv6 procedures for using Xcast in a domain*

The MAG does not change its behavior and subscribes to the multicast content on behalf of the MNs (acting as a proxy) by using a multicast group membership protocol such as MLD. The multicast content requests will reach the mobility anchor through the tunnel, following the standard IP-in-IP encapsulation [1]. The mobility anchor will act as an Xcast source, and will take the decision of encapsulating the multicast traffic in an IP-in-Xcast mode in its downstream interfaces reaching the MAGs, instead of using the standard IP-in-IP tunnel.

The router present in the bifurcation point in the end to end path providing connectivity the last segment to reach a MAG (i.e., no more MAGs reachable through that branch from that router), will send the multicast packet in unicast fashion as in the IP-in-IP case (see Figure 2), so the MAG will not perceived any change in the multicast distribution regarding the standard case.

Two ways of Xcast distribution can be considered. On one hand, it can be considered that all the subscriptions between a set of MAGs and the mobility anchor are distributed over the same IP-in-Xcast tunnel, grouping all the multicast channels subscribed for a certain group of MAGs. On the other hand, it can be considered that a separate IP-in-Xcast tunnel is used per multicast channel.

The tunnel management is very complex in the first case, as the tunnel has to be dynamically updated. Furthermore, different subscription groupings should be arranged according to the subscriptions existing on the MAGs. The second case is simpler, and it is the one selected in this paper.



(a) (b)

Figure 4. Exampleof *k-tree* structure with *k*=2 and *D*=4 (a), and parameters defining a a *k-tree* structure (b).

D. *Handover dynamics using Xcast*

In the event of a handover, the movement of an MN can produce the need of creating a new branch for the distribution of the multicast content to the MAG where the MN is being attached (in case there is no other MN in that MAG subscribed to the desired content). Similarly, such movement can produce the removal of an existing branch from the MAG where previously it was attached (in case the MN was the last subscriber to a certain content).

As the LMA is aware of the channels subscribed per MAG in the remote subscription case, the LMA has to take the decision of Xcast forwarding to the MAG. When a new branch has to be formed due to a handover event, it will simply mean the need of adding a new destination to the Xcast header pointing to the requesting MAG. Furthermore, when an existing branch has to be removed also for a handover event, the LMA has just to remove the corresponding MAG from the desired destinations from the Xcast header.

IV. PERFORMANCE COMPARISON

A. *Definition of the scenarios under analysis*

In order to evaluate and compare the potential gains in the use of Xcast for transporting the multicast traffic between the mobility anchor and the access gateways within a PMIPv6 domain, we will model the distribution tree with a *k-tree* structure as considered in [6]. Figure 4 (a) shows an example of a *k-tree* composed by a total number of 31 nodes (i.e., *k*=2, *D*=4).

A *k-tree* structure can be characterized by two parameters, as depicted in Figure 4 (b): *k*, the degree of the tree or number of leaves recursively found from every previous leaf on the tree, and *D*, the depth of the tree, which indicates the number of levels in the distribution tree.

N, the total number of nodes in a certain *k-tree*, is given by:

$$N = \frac{k^{D+1} - 1}{k - 1}, \tag{1}$$

while the number of potential receivers (MAGs in this analysis), *m*, is obtained from:

$$m = k^{D} \tag{2}$$

B. *Performance evaluation*

The performance evaluation of Xcast versus the standard distribution is carried out by comparing the number of traversed links between the mobility anchor and the access gateways when each of these solutions is used to distribute a multicast channel.

1) General calculation

The previous calculation has considered that all MAGs subscribe to the same content. While this can be true for highly demanded content, it cannot be generalized. In this section we try to formulate the generic calculation of the links traversed in a PMIPv6 as a function of the demand.

A certain channel will be subscribed by the MAG if there is at least one attached MN demanding such channel. Let us consider p as the probability of a MAG demanding a certain channel, then it can be established that $p=1$ if there is at least an MN requesting the channel, and $p=0$ otherwise.

Then, in the standard case, the links traversed for serving the MAGs demanding the channel will be:

$$L_{MAG}^{std} = p \times D \,. \tag{3}$$

The total number of links traversed in the PMIPv6 domain can be established on:

$$L_{Total}^{std} = \sum_{i=1}^{m} p_i \times D = D \times \sum_{i=1}^{k^D} p_i \tag{4}$$

For the Xcast case, it is a bit more complex to calculate the link usage, as the usage at a level of the tree depends on the usage of the following level, as can be derived from Figure 4. A link of a certain level will not be traversed if none of the receivers (MAGs) below it subscribes to such content. This can be formulated in the following way:

$$L_1^{Total} = \sum_{i=1}^{m} (1-(1-p_i)) = \sum_{i=1}^{k^D} (1-(1-p_i)) \tag{5}$$

$$L_2^{Total} = \sum_{i=1}^{m/k} (1-(1-p_i)^k) = \sum_{i=1}^{k^{D-1}} (1-(1-p_i)^k)\,, \tag{6}$$

and, in general:

$$L_d^{Total} = \sum_{i=1}^{m/k^{(d-1)}} (1-(1-p_i)^{k^{(d-1)}}) = \sum_{i=1}^{k^{D-(d-1)}} (1-(1-p_i)^{k^{(d-1)}})\,. \tag{7}$$

Then, the total number of links traversed in the Xcast case will be the sum of the links used for all the levels, given by:

$$L_{Total}^{Xcast} = L_1^{Total} + L_2^{Total} + \ldots + L_D^{Total} \tag{8}$$

Figures 5 and 6 show the comparison of the gain obtained for different *k-trees* distribution architectures connecting the same number of mobile access gateways, as a function of the probability of subscription p for a certain multicast channel per MAG (note that the probability of subscription of any MAG is independent of the probability of subscription of the other MAGs in the domain, then it can be stated that $p_i = p,\ \forall i$, assuming a similar average number of MNs per MAG). In the first figure, 64 MAGs are connected through two different *k-tree* structures, with 8 degrees and 2 depth levels in one case, and 4 degrees and 3 depth levels in the second. The number of intermediate nodes changes, obtaining in the first case a flatter architecture.

The second figure, considering 729 MAGs (which can be connected either by *k-trees* of parameters k=9 and D=3, or k=3 and D=6), is presented to evaluate the sensitivity of Xcast to the growth in the number of connected MAGs, only for illustrative purposes, as such high number of MAGs cannot be included in a unique Xcast tree due to the limitation on the number of destinations per Xcast header.

As observed from Figures 5 and 6, more hierarchical *k-tree* structures provide more savings that their flattened counterparts for connecting the same number of MAGs. This trend is more significant as the number of MAGs in the tree increases. The gain increases rapidly with the probability of subscription per MAG p, and it is asymptotically bounded, which means that the maximum network resource savings are closely reached even for moderately popular channels. Finally, it can be concluded that higher savings are obtained as the number of MAGs grows in the domain.

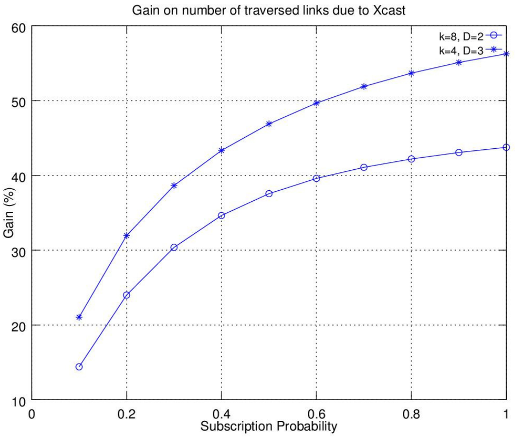


Figure 5. Gain due to multicast for two different *k-tree* structures connecting 64 MAGs

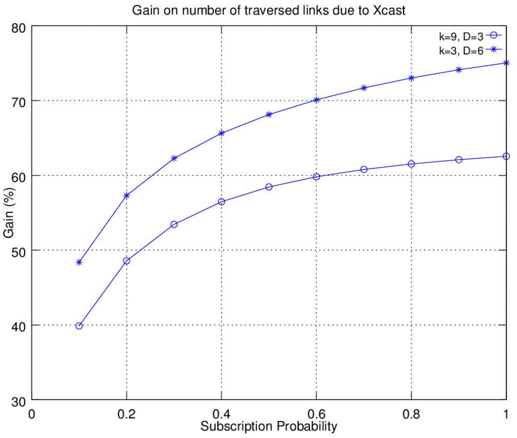


Figure 6. Gain due to multicast for two different *k-tree* structures connecting 729 MAGs (for illustrative purposes)

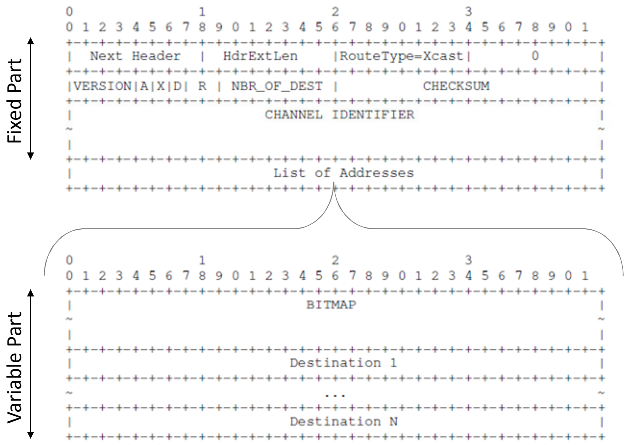


Figure 7. Xcast header format

C. *Overhead calculation*

Another relevant aspect is the comparison between multicast and Xcast in terms of the total overhead on the distribution structure under analysis. The standard distribution of the remotely subscribed multicast context uses IP-in-IP encapsulation, and therefore the overhead is due to the IP encapsulation in the bi-directional tunnel. In our proposal, the remotely subscribed multicast content is distributed in an IP-in-Xcast fashion; then the overhead will be originated by the Xcast mechanism.

1) Standard multicast case

Considering h_{IP} as the overhead bits needed for the encapsulation of the remote multicast channel (i.e. the 40 bytes of an IPv6 header in an IP-in-IP tunnel), the total overhead due for transporting a multicast channel all the path from the mobility anchor to an access gateway can be stated as:

$$O_{ch-MAG}^{std} = D \times h_{IP} \cdot \tag{9}$$

Extending this formula to the set of MAGs in the domain, the total overhead for a multicast channel being distributed to all the MAGs across the PMIPv6 domain can be written as:

$$O_{ch-domain}^{std} = k^D \times D \times h_{IP} \cdot \tag{10}$$

2) Xcast case

In Xcast definition, the encapsulation defined is composed of an IPv6 header and an Xcast header, carried as a routing extension, which is structured in a fixed part and a variable one.

The IPv6 header will have as source address the address of the Xcast sender (the mobility anchor in our case), being the destination address the "all_Xcast_routers" address. As consequence of the IPv6 header, every Xcast packet will account h_{IP} bytes.

The Xcast header presents a fixed 24-byte part including several protocol fields. Among them, the NBR_OF_DEST field determines the maximum number of destinations that can be included in the Xcast header. This field is 7 bit long, so a maximum of 127 destinations could be included in an Xcast distribution. The issue on the number of the maximum number of the destinations (i.e., MAGs) is discussed on Appendix A, at the end of the paper.

The variable part of the Xcast header will carry the list of the destination addresses for packet forwarding. Each Xcast router in the path will evaluate the list of destinations to replicate the packet accordingly for each of the corresponding next hops, including on the next packet just the destinations to be routed through the next hop, onwards. This variable part includes also a BITMAP field, of which size depends on the number of destinations, being a multiple of 64 bits.

Then, the size of an Xcast header for a certain distribution level in the *k-tree* can be can be formulated as:

$$h_{xcast}^{L_d}(bytes) = 24 + N_{L_d} \times 16 + \left\lceil \frac{N_{L_d}}{64} \right\rceil \times 8 , \tag{11}$$

being N_{Ld} the number of destinations reachable from level L_d in a certain branch, that can be defined in the following manner:

$$N_{L_d} = k^{(d-1)} \cdot \tag{12}$$

Each Xcast packet is converted into a normal unicast packet for reaching the last destination. In that case, corresponding to the first level in the tree, L_1, the applicable overhead will be just h_{IP}, with $h_{xcast}^{L_1} = 0$.

When extending these formulas to the whole set of MAGs in the domain, we obtain:

$$O_{ch-L_1}^{xcast} = k^D \times (h_{IP} + h_{xcast}^{L_1}) = k^D \times h_{IP} \tag{13}$$

$$\begin{aligned} O_{ch-L_2}^{xcast} &= k^{D-1} \times \left(h_{IP} + h_{xcast}^{L_2}\right) = k^{D-1} \times h_{IP} + \\ &+ k^{D-1} \times \left(24 + k \times 16 + \left\lceil \frac{k}{64} \right\rceil \times 8\right) \end{aligned} \tag{14}$$

$$\begin{aligned} O_{ch-L_3}^{xcast} &= k^{D-2} \times \left(h_{IP} + h_{xcast}^{L_3}\right) = k^{D-2} \times h_{IP} + \\ &+ k^{D-2} \times \left(24 + k^2 \times 16 + \left\lceil \frac{k^2}{64} \right\rceil \times 8\right), \end{aligned} \tag{15}$$

and, in general:

$$\begin{aligned} O_{ch-L_d}^{xcast} &= k^{D-(d-1)} \times \left(h_{IP} + h_{xcast}^{L_d}\right) = k^{D-(d-1)} \times h_{IP} + \\ &+ k^{D-(d-1)} \times \left(24 + k^{d-1} \times 16 + \left\lceil \frac{k^{d-1}}{64} \right\rceil \times 8\right) \end{aligned} \tag{16}$$

Then, considering the total distribution in the PMIPv6 domain, the general formulation of the overhead required for distributing a multicast channel to all the MAGs is given by:

$$\begin{aligned} O^{xcast}_{ch-domain} &= \sum_{d=1}^{D} O^{xcast}_{ch-L_d} = \\ &= \sum_{d=1}^{D} k^{D-(d-1)} \times h_{IP} + \sum_{d=2}^{D} k^{D-(d-1)} \times h^{L_d}_{xcast}, \end{aligned} \tag{17}$$

that can be rewritten as:

$$O^{xcast}_{ch-domain} = k\frac{(k^{D}-1)}{k-1} \times h_{IP} + \sum_{d=2}^{D} k^{D-(d-1)} \times h^{L_d}_{xcast}. \tag{18}$$

Figures 8 and 9 present a comparison of the overhead for distributing a channel to all the MAGs in a domain, considering different *k-tree* configurations.

The Xcast option introduces less overhead than the standard case as the degree in the tree, *k*, grows for a given tree depth, *D*. At the same time, as the depth of the tree *D* increases, the advantage on using Xcast becomes more significant for higher *k-tree* degrees.

V. CONCLUSIONS AND FURTHER WORK

As shown in the previous analysis, the Xcast encapsulation can provide a lower cost per transported bit for a PMIPv6 domain operator offering remote multicast distribution capabilities to a Home Network operator, allowing for better benefit margins. It can also be concluded that the most efficient distribution structures for serving a certain number of MAGs in the PMIPv6 domain are those more hierarchical (i.e., with greater number of levels, *D*), instead of the flatten ones, because a higher gain is achieved respect to the standard multicast case. This matches existing operators' network topologies. Furthermore, higher degrees in the tree result in less overhead for the Xcast case.

As next steps, we are working on the characterization of the total overhead as a function of the channel subscription probability at the MAG. We are also studying how to dynamically decide when to use standard multicast versus Xcast transport depending on the locations of the MAGs subscribing the content in the *k-tree*, and in general, depending on the number of MAGs subscribing the content, for alleviating intermediate routers of the burden of Xcast processing in scenarios of low gain.

ACKNOWLEDGMENTS

The research of Carlos J. Bernardos leading to these results has received funding from the European's Community's Seventh Framework Programme (FP7-ICT-2009-5) under the grant agreement n. 258053 (MEDIEVAL project), being also partially supported by the Ministry of Science and Innovation (MICINN) of Spain under the QUARTET project (TIN2009-13992-C02-01).

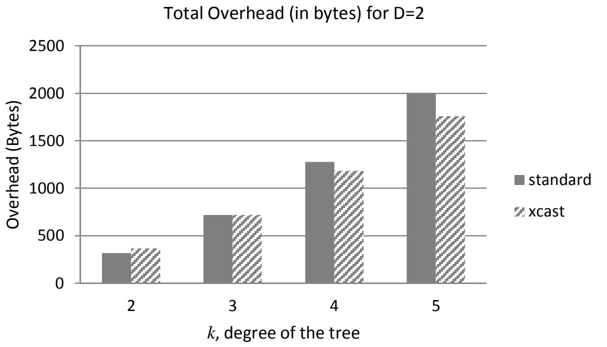


Figure 8. Total overhead comparison for different degrees values in a *k-tree* with depth *D*=2

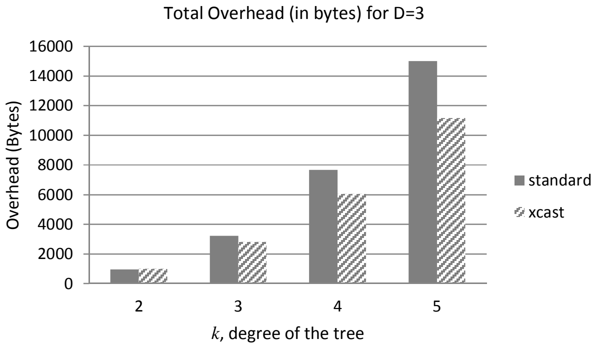


Figure 9. Total overhead comparison for different degrees values in a *k-tree* with depth *D*=3

REFERENCES

[1] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, B. Patil, "Proxy Mobile IPv6", RFC 5213, August, 2008.

[2] 3GPP, "Architecture enhancements for non-3GPP accesses", 3GPP TS 23.402 10.5.0, September 2011.

[3] T.C. Schmidt, M. Waehlisch, and S. Krishnan, "A Minimal Deployment Option for Multicast Listeners in PMIPv6 Domains", RFC 6224, April, 2011.

[4] R. Boivie, N. Feldman, Y. Imai, W. Livens, D. Ooms, "Explicit multicast (Xcast) concepts and options", RFC 5058, November, 2007.

[5] J.C. Zuniga, L.M. Contreras, C.J. Bernardos, S. Jeon, Y. Kim, "Multicast mobility routing optimizations for Proxy Mobile IPv6", RFC 7028, September, 2013.

[6] P. Van Mieghem, G. Hooghiemstra, R. van der Hofstad, "On the efficiency of multicast", IEEE/ACM Transactions on Networking, Vol. 9, No. 6, pp. 719-732, December, 2001.

[7] H. Luo, H. Zhang, Y. Qin, V.C.M. Leung, "An approach for building scalable Proxy Mobile IPv6 domains", IEEE Transactions on Network and Service Management, Vol. 8, No.3, pp. 176-189, September, 2011.

[8] C. Perkins, D. Johnson, J. Arkko, "Mobility support in IPv6", RFC 6275, July 2011.

[9] Cisco 8500 Series Wireless Controller Deployment Guide, Document ID 113695, October 2012 (downloadable at: http://www.cisco.com/image/gif/paws/113695/8500-dg-00.pdf), accessed in November 2012.

APPENDIX A – ON THE NUMBER OF MAGS IN A PMIPV6 DOMAIN

This Appendix aims at discussing the typical number of MAGs in a PMIPv6 domain as a way of determining the potential limitations on the use of Xcast in a PMIPv6 domain.

The length of the field NBR_OF_DEST (7 bit) of the Xcast header limits the maximum of destinations to 127. Note that this limit applies to each branch of the *k-tree*, in such a way that the total number of MAGs per *k-tree* (i.e., per mobility anchor) can be raised to $k \times 127$. In order to evaluate how this number could be in line with the number of MAGs in a PMIPv6 deployment, we will analyze the potential scalability of a PMIPv6 domain in terms of users and number of access gateways.

We next partially follow the analysis considered in [7]. There, authors looked at the bandwidth requirements of the mobility anchor as one of the limiting factors for this entity. Being R_{OS} the rate of oversubscription in the LMA (that is, the rate of the total number of MNs registered in excess regarding the number of actually active MNs), and T_p the peak data throughput per active MN, the bandwidth delivered by the mobility anchor equals[1] to:

$$BW_{anchor} = M \times \frac{T_p}{\left(R_{OS} + 1\right)}, \tag{19}$$

where M represents the total number of MNs registered at the mobility anchor in the PMIPv6 domain.

Commercial off-the-self core routers today are capable of delivering traffic in the order of Tbps. Table II summarizes the achievable number of MNs in the PMIPv6 domain assuming a mobility anchor forwarding capacity of 1 Tbps, considering different values of the observable peak data throughput and oversubscription ratios.

TABLE I. NUMBER OF REGISTERED MNS PER MOBILITY ANCHOR WITH A FORWARDING CAPACITY OF 1 TBPS

	Peak data throughput, T_p		
Oversubscr., R_{OS}	100kbps	1Mbps	10Mbps
0	10^7	10^6	10^5
5	6×10^7	6×10^6	6×10^5
10	11×10^7	11×10^6	11×10^5

For every attached MN, the mobility anchor has to keep an entry in the binding cache. Such an entry contains a number of fields [1] [8], like the MN identifier (128 bits), the MN's link layer identifier (64 bits), the MAG's link layer identifier (128 bits), the list of the Home Network Prefixes (HNPs) for the MN's interface (each prefix being 128 bits), the tunnel identifier (at most 128 bits), the Proxy Care-of-Address (128 bits), etc. These fields require a storage capacity above 1000 bits per MN. Taken this into account, the upper limit in the number of MNs observed in Table II imposes the need of handling a global binding cache memory in the order of 10 Tbits. This huge storage capacity, the corresponding number of associated routing entries, and the lookup capacity required to handle both of them, make that upper limit unachievable. Therefore, we can argue that a more realistic upper limit of MNs managed by a mobility anchor would be in the order of few hundreds of thousand terminals.

Current state-of-the-art MAG specifications [9] support a maximum number of 40,000 attached MNs. This implies that only a branch of the *k-tree* (that is, a downstream interface of the mobility anchor), with 127 of those MAGs, could potentially provide connectivity to more than 5 million MNs, much more than the total number of MNs per mobility anchor. To sum up, it can be stated that the number of MAGs per branch will be lower than the limit imposed by the field NBR_OF_DEST in the Xcast header.

---

[1] We have slightly modified the formula used for obtaining Fig. 19 in [7] because it is not totally correct from our point of view. In the original formula the denominator only considers the rate of oversubscription, R_{OS}, not providing a consistent result for the case when no oversubscription occurs.