

# Multiparty Seamless Transport

Nuno Coutinho<sup>1</sup>, Rui Valbom<sup>1</sup>, Tiago Condeixa<sup>1</sup>, Susana Sargento<sup>1</sup>, Augusto Neto<sup>1 2</sup>

<sup>1</sup> Instituto de Telecomunicações, University of Aveiro, Portugal

{nunocoutinho,rui.valbom,tcondeixa,susana}@ua.pt, augusto@av.it.pt

<sup>2</sup> Instituto de Informática, Universidade Federal de Goiás, Brazil  
augusto@inf.ufg.br

**Abstract**—The demanding requirements of new real-time and mobile multimedia group-based applications (e.g. IPTV and video-conferencing) motivates the research for novel and efficient network architectures, able to fulfil service requirements and increase user experience. Multicast, despite its scalability issues, is the most suitable network technology for the delivery of multiparty content. Context-awareness is taken as a key feature in Next Generation Networks, enabling a management framework with a more comprehensive knowledge of the network, application and user. The challenge is how to use the available knowledge towards service personalization and network efficiency. This paper presents and evaluates an architecture that aims to provide personalized session content delivery to multiple mobile users, integrating different access and transport technologies. The architecture relies on an end-to-end context-aware communication framework able to dynamically react and adapt to context changes, through session and network reconfiguration. Our focus is on the transport network part, addressing a new level of scalable multiparty content delivery through the concepts of abstract multiparty trees and overlay nodes. The results show that the concept of abstract trees reduces the network reconfiguration complexity while still guaranteeing end-to-end QoS.

**Index Terms**—Multicast, Quality of Service, Overlay, Abstract Multiparty Trees, Context-awareness, Heterogeneity, Scalability.

## I. INTRODUCTION

Internet services and telecommunication systems are always evolving towards efficient service provision and customer satisfaction. The latest can be achieved by enriching services with novel approaches, as context-awareness, mobility support and ability to adapt to environment changes, offering a new level of personalized communication. In this sense, and given the increasing demand for group-based multimedia (e.g. IPTV) services, context information can be used towards an adaptive and personalized multiparty content delivery. This context-awareness raises networking complexity, due to the amount of information involved, constant context changes and mobility. Thus, the development of scalable optimization schemes are required to support context-aware content delivery under heterogeneous and dynamic conditions.

This paper introduces a future network architecture capable of supporting context-aware content delivery to multiple mobile users, independent of network and access technologies. Through the perception of the features of the communication environment, we expect to take advantage of context heterogeneity in order to better adapt the communication to the current network and surroundings conditions. Moreover, this knowledge may also be employed towards the provision

of more personalized services, meeting user capabilities and preferences. However, as mobile applications are becoming more demanding, QoS cannot be neglected while providing group services. Considering the dynamics of future networking environments, constant network adaptations and reconfigurations will be required according to users, session and network context changes. Hence, network control must be scalable to address this added complexity. In this sense, we developed an hierarchical strategy, using several functional components driven by context to support intelligent decisions and adaptations to optimize content delivery. We develop the concept of abstract trees that add greater flexibility in the control of multicast distribution trees, hiding network dynamics through internal local reconfigurations. The results of the proposed context-aware multiparty transport framework show that this hierarchical architecture is able to reduce the complexity of the reconfiguration processes while still achieving the required quality of the services.

The remainder of this paper is organized as follows. The related work is briefly presented in Section II. Section III describes the architecture, functions and mechanisms. Section IV presents the evaluation of the architecture. Finally, Section V concludes the paper and introduces the future work.

## II. RELATED WORK

Considering group-based applications, multicast is the most suitable technology to save network resources. However, its open service model and scalability issues [1] detained its commercial deployment. Proposals as the Explicit Multicast [2] and the Hop by Hop Multicast Routing protocol (HBH) [3] are focused on improving the scalability and the interoperability between unicast and multicast. Novel strategies for the progressive deployment of multicast have also gained increasing prominence, as the Application Layer Multicast (ALM) [4]. ALM solutions implement multicasting functionality at the application layer, making possible to overcome infrastructure deployment issues at the cost of non-optimal distribution trees.

These developments usually do not address QoS control mechanisms. However, some research has been made under this topic, as the integration of class-based QoS and IP multicast [5]. In the DiffServ model [6], the scalability is achieved by pushing complexity to network edges and maintaining per-class state in core nodes, while multicast operates on a per-flow basis through the network. A current problem that must be considered is related with routing asymmetries [7]. However, this issue is not tackled by most IP multicast routing protocols,

since these protocols build multicast trees from the receivers to the sender, while data travels in the reverse direction. The *QoS Architecture for Multi-user Mobile Multimedia (Q3M)* [8] is one of the works that addresses this issue, introducing the Multi-service Resource Allocation (MIRA)[9] protocol, designed to control the resources of the CoS and multicast trees in a synchronized manner taking into account the problem of asymmetric routes. A centralized structure to support QoS was developed in [10], enabling QoS and mobility of multicast sessions in heterogeneous environments. More recently, new algorithms and techniques are proposed, such as the QoS mobile multicast routing based on ant colony optimization [11], which has a strong global optimization performance.

Beyond these works, there is a common vision in the related work that context-awareness is crucial for future network architectures, endowing them with dynamic adaptability to user, network and environment features. Context, as defined by [12], is any information that can be used to characterize an entity. Thus, it is essential to develop frameworks to gather, manage and disseminate contextual information, as in [13][14][15]. The knowledge acquired can then be used to enrich tree construction algorithms and QoS mechanisms.

### III. CONTEXT-AWARE MULTIPARTY TRANSPORT

This paper proposes a novel framework to enable personalized and context-aware content delivery to multiple mobile users, regardless of the network access and transport technologies. This multiparty transport approach takes advantage of components developed at different levels, Fig. 1: *i)* Context Detection and Distribution framework; *ii)* Multiparty Session Management; *iii)* Multiparty Transport Framework.

The Context-awareness framework is composed by sensors responsible for acquiring context information about network, users/terminals and surroundings to feed a context database and reasoner, the Context Broker (CB). This framework provides the knowledge to optimize the content delivery, reacting and adapting to new networking conditions. The Multiparty Session Management works as an overlay between applications and networks, handling session events; it creates the session context, consisting of session's group members identification and QoS requirements. The novel feature of this functional block is performed by the Group Manager element, which groups or sub-groups members taking into account session context and users' preferences, device capabilities and environment context. This feature aims to provide a more personalized content delivery in group communications, where sub-groups may use different media encodings of the same content, allowing a better match between user context and current network conditions.

This paper proposes a Context-Aware Multiparty Transport Framework composed by a set of elements that allows the use of the knowledge gathered and reasoned in operations such as network selection, grouping users and resource control. The aim of these elements is to support an efficient and scalable network adaptation to any context changes, through a hierarchical approach for network control, providing a flexible

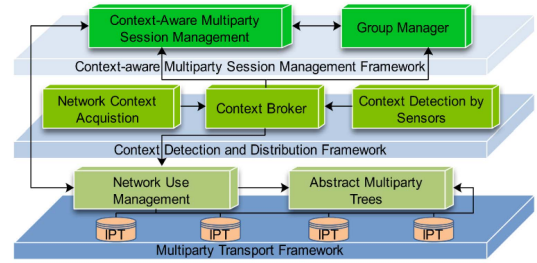


Figure 1. C-CAST functional architecture

resource allocation mechanism able to offer multicast services with delivery guarantees.

#### A. Hierarchical Network Control

Since involving more information in the management and control processes increases the complexity, scalability becomes a main concern in the architecture design. Our aim is to provide a control framework able to react in a scalable way to any context change without impairing the services quality. In this sense, a hierarchical control framework is proposed in this paper, aiming to employ two levels of intelligence in the network. The higher level consists on a central intelligence entity, named Network Use Management (NUM), which makes use of context information available to proper react to context changes or events. The next level of network intelligence consists on providing network decision functionalities to the edges of the abstract multiparty trees (AMTs), the Overlay Nodes (ON). The AMTs provide a generic and scalable transport service for group communications. Each ON performs resource and QoS control in a distributed manner through the Internet Protocol Transport (IPT) components present in each network router.

Before detailing the features of the above mentioned components, we first include a formal definition of the network and the overlay structure used. We consider a network formally defined as  $G = (V, E)$ , where  $V = \{v_1, v_2, \dots, v_{|V|}\}$  represents the set of nodes.  $E$  is the link set, where  $e = (v_a, v_b) \in E$  represents the link connecting  $v_a$  to  $v_b$ . Among the available network nodes ( $V$ ), there is a subset of nodes  $V_O \subset V$  that can perform overlay functions and being edges of AMTs. Thus, considering the substrate network  $G$  we define  $O = (V_O, E_O)$  as the overlay network, where  $E_O$  is the set of logical links and  $e_O = (v_{Oa}, v_{Ob})$  represents the overlay connection. Moreover, an overlay connection between two ONs may be mapped into a set of physical network links  $e_O = \{e_1, e_2, \dots, e_l\}$ , being  $l$  the number of physical links in an overlay connection. We also define the set of existent multicast sessions as  $M$ , where each one has associated a content source  $S_{M_i}$  and a set of flows  $F_{M_i} = \{F_1, F_2, \dots, F_{|F_i|}\}$ . Each flow has a set of receivers  $R_{F_i}$  and a multicast tree  $G_{F_i}$ . The available set of access points (APs) in the network is defined as  $Z = \{z_1, z_2, \dots, z_{|z|}\}$ , having each receiver an associated access point  $z_j$ .

#### B. Abstract Multiparty Transport

The concept of Abstracted Multicast Tree (AMT) resides in defining an overlay structure  $O_{F_i} \subset O$  for multicast session

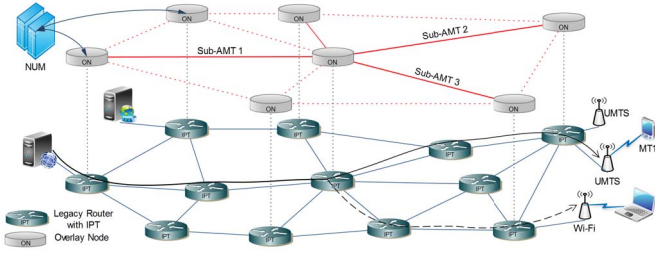


Figure 2. End-to-end AMTs and interiors Sub-AMTs.

flows  $F_i$ , operating on top of the IP network layer to support general transport control in multicast trees, Fig. 2. By applying this overlay paradigm at the transport layer, it is possible to provide a scalable transport service for group communications. The end-to-end AMTs will then be divided in sub-AMTs, generated between the ONs, that will split the end-to-end multicast tree. This abstraction level and division of trees enables end-to-end multicast content transport over heterogeneous network segments in terms of IP multicast capability or IPv4/v6 support (each sub-AMT has associated a unique multicast address), also hiding network dynamics (the reconfigurations inside each sub-AMT can be performed locally to the sub-AMT), and providing independence between source and listeners (several sub-AMTs are supported independently).

The local network segments are called sub-AMTs, and they may be defined as an overlay connection ( $e_O$ ) between two adjacent ONs ( $v_{Oa}, v_{Ob}$ ). All core nodes composing a sub-AMT must implement the same transport technology. In the scope of a sub-AMT: (i) ingress ON,  $v_{Oa}$ , is viewed as a session source; (ii) egress ON,  $v_{Ob}$ , as leaf node; (iii) and core nodes simply perform IP forwarding operations.

NUM will coordinate the edges of each sub-AMT for mediating overlay connections, so ONs may control network resources and QoS of each sub-AMT independently from the others. A detailed description of NUM features will be given in the following section.

### C. Network Use Management

NUM represents the higher level of architecture intelligence, being responsible for the control of the session setup procedure and the management of the overall network rearrangements based on context information. Network selection is one of NUM's features, being divided into access and core network selection. The former is essential given the heterogeneous and multihomed environments of NGN, attempting to connect terminals to the access networks that best suit their communication needs. To accomplish this, NUM comprises users' context, session requirements and network context in its access selection mechanism. As each access is selected, it is required to determine which nodes are ONs to build the sub-AMTs from source to destinations, according to the resources available in the links between ONs and the requirements of the flows. This is the objective of the (*CoreSelection* function).

The functionalities performed by NUM are detailed on Algorithm 1, which can be triggered by two main events: a new

### Algorithm 1: NUM Functionalities

```

input :  $G, M_i, S_{M_i}, F_{M_i}, R_{F_i}, Z, O_F = Empty$ 
output: An AMT for each Flow:  $O_{F_i}, i = 1 \rightarrow |F|$ 

if New  $M_i$  Request then
    Request Context from CB;
    for  $i = 1 \rightarrow |F|$  do
        for  $j = 1 \rightarrow |R_{F_i}|$  do
             $z_{R_j} = AccessSelection(R_j, F_i, G, Z);$ 
             $O_{F_i} = CoreSelection(O_{F_{i-1}}, S_{M_i}, F_i, G, z_{R_j});$ 
        end
    end
else if Ongoing  $M_i$  then
    switch Event do
        case Core event
             $O_{F_i} = CoreSelection(O_{F_i}, F_i, G, z);$ 
        case Access event
             $z_{R_j} = AccessSelection(R_j, F_i, G, Z);$ 
            if  $z_{R_j}$  is connected to a new  $v_O$  then
                 $O_{F_i} = CoreSelection(O_{F_i}, F_i, G, z_{R_j});$ 
            end
        endsw
    endsw
end
    
```

multicast session request, sent by the Session Management component, or an event that requires network rearrangement. NUM receives from the Session Management information about Multicast Session ( $M_i$ ), Session Source ( $S_{M_i}$ ), Session Flows ( $F_{M_i}$ ) and Receivers associated to each Flow ( $R_{F_i}$ ). The remaining required information is obtained from the context broker, as network knowledge ( $G$ ), APs available ( $Z$ ) and users context information. In case of a new multicast session request, NUM will select the most suitable AP according to user and network context, session lect the several ONS to build the sub-AMTs. The algorithms employed in each of the network selection functions (access and core) are presented in [16].

In case of being an event that triggered NUM, different reaction strategies may be applied according to the cause of the event. If the event is a link/node failure or QoS degradation, the recovery can be done internally to the sub-AMTs through the ONs, or the recovery needs a change on the sub-AMTs and then it is required to perform a new core selection procedure to overcome the failure. On the other hand, if the event's cause is on the access network part, beyond the always necessary new access selection, it may also be required a new core selection if the selected AP is connected to a new ON ( $v_O$ ). The output of NUM will always be a set of sub-AMTs  $O_{F_i}$  to control the multicast tree and the resources for a specific flow  $F_i$ .

### D. Overlay Node Functions and Internet Protocol Transport

ONs are in charge of setting up sub-AMTs to which they are individually associated, controlling their resources and QoS through the IPT elements within the scope of the sub-AMT. They represent the second level of network intelligence which, despite being controlled and managed by NUM, present autonomous mechanisms to manage their sub-AMTs. The functionalities performed by an ON and IPTs within its domain are detailed on Algorithm 2. Each ON receives from NUM the information about the sub-AMT to which it is associated, as well as session requirements (bandwidth, delay). Regarding ON functionalities, two different procedures may occur depending on the trigger event.

**Algorithm 2: ON and IPT Functionalities**


---

```

input :  $G, M_i, S_{M_i}, F_{M_i}, R_{F_i}, Z, O_F = Empty$ 
output: An AMT for each Flow:  $O_{F_i}, i = 1 \rightarrow |F|$ 
switch Trigger Event do
  case Sub-AMT Setup Request
    Allocate IP Multicast Tree;
    repeat For each IPT
      Resource Allocation according to session CoS;
      Join IP multicast tree;
      Update MRIB;
    until Next ON;
    Configure ON Proxy Settings;
  endsw
  case Sub-AMT Rearrangement
    Check RIB for route alternatives until next ON;
    if Successful then Local Rerouting;
    else Notify NUM for global rearrangement;
  endsw
endsw

```

---

In case of a new *Sub-AMT Setup Request*, the ON starts by assigning a multicast address to the tree. Then, for each IPT within the physical path of the sub-AMT (until reaching the other edge ON), the ingress ON will trigger network resource allocation according to the session requirements (bandwidth, delay, etc.) and then performs the router *Join* process to the multicast tree. Among the IPT enforcing tasks, it is also present the update of the multicast routing information table (MRIB), in order to employ, throughout the physical links and nodes, the route in which the sub-AMT is mapped. Finally, and regarding the different multicast addresses assigned to each sub-AMT, proxy settings need to be configured in the ONs. Thus, ONs change the destination IP address of packets that belong to the flow associated with the sub-AMT configured, updating them with the multicast address of the sub-AMT.

In case of a *Sub-AMT Rearrangement*, ONs have autonomous features that make them able to rearrange their controlled sub-AMTs without notifying NUM. To achieve this, in case of link/node failure or QoS degradation, the ingress ON of a certain sub-AMT checks in the Routing Information Base (RIB) for routing alternatives with the same characteristics to the egress ON. If a new route is available, the ON has the capability to update the MRIB and enforce the new path, rearranging the sub-AMT without NUM's knowledge. Otherwise, NUM has to be notified and select a new egress ON for the sub-AMT.

### E. Network Reconfiguration

The Context-aware Multiparty Transport framework presented in this paper aims to efficiently support the dynamics imposed by the constant adaptations to context changes. Comparing with MIRA [9], this approach endows the global AMT with more flexibility, since it is possible to reconfigure only part of it (Sub-AMTs) and not all the multicast distribution tree. Following MIRA strategy it would be necessary to reconfigure the entire path from the ingress router until the egress router. Thus, the concept of AMTs described introduces in the architecture a new level of scalability avoiding global network reconfigurations

For a better understanding of the flexibility added by the AMT concept, we consider an AMT associated to a flow  $F$

formally defined as  $O_F = \{V_O, E_O\}$ . Assuming that an event triggers a network reconfiguration, the outcome of this process is an updated version of the previous AMT,  $O'_F = \{V'_O, E'_O\}$ . From a theoretical perspective, the advantages of employing abstract transport and of reducing the impact on the network depends on the amount of links and configured resources that can be reused by the new overlay structure  $O'_F$ . Thus, the efficiency of the AMT concept depends on various factors: the ratio between overlay capable nodes and normal nodes,  $\lambda = \frac{V'_O}{V}$ ; and the average number of physical links ( $\bar{l}$ ) that maps an overlay connection ( $e_O = \{e_1, e_2, \dots, e_l\}$ ).

Considering this, and defining the previous AMT as  $O_F$  and the new one as  $O'_F$ , the number of reused overlay nodes is given by  $r = |V_O \cap V'_O|$ . We may then assume that  $r \propto \lambda$ , meaning that, increasing the percentage of ONs in the network will decrease the number of configurations required, since part of the overlay is reused. Thus, it is possible to diminish the amount of control overhead and only perform the resource reservation for the new branch of the AMT. However, this fact does not mean that a network full of overlays would be the best solution. Since the control architecture is hierarchical, there is a trade-off between the gains obtained by reusing previously allocated overlay connections and the amount of control that NUM has to perform in order to manage the overlay network.

Moreover, it is implied that  $\lambda \propto \frac{1}{\bar{l}}$ , which reduces the physical range of each overlay connection (sub-AMT). As ONs are endowed with the ability to perform some autonomous functions within the scope of their associated sub-AMTs ( $e_O$ ), e.g. re-routing and resource allocation (Algorithm 2), increasing their number will cause a loss of the self-configuration ability of the architecture.

Considering a source specific multicast protocol as PIM-SSM [17], and despite not addressing QoS issues, it tends to have a "selfish" behavior concerning the multicast tree construction. To build its multicast distribution tree, PIM-SSM uses the unicast routing tables to forward *Join* messages until it finds the respective source tree. Thus, considering receivers of the same group sparsely located in the network, it is likely that each one has an associated tree whose set of nodes and links is disjoint from another receiver's source tree. In this case, a tree rearrangement triggered by any kind of event would require a complete setup of the path between the receiver and the source, being unaware of tree branches that could be reused. Using the AMT strategy with the control associated, we attempt to aggregate the content distribution as much as possible, while providing QoS guarantees.

## IV. EVALUATION

The architecture described so far was implemented in the Network Simulator 2 (NS-2) [18]. In order to assess its performance and robustness, it was created a flexible network topology regarding the number of ingress, egress and core nodes, core ONs, APs, mobile terminals (MTs), data sources and sessions. This way, we easily evaluate the response of the implemented scheme by varying the input parameters and generating different scenarios. All experiments were based on the scenario of Fig. 3, with 9 network nodes, changing

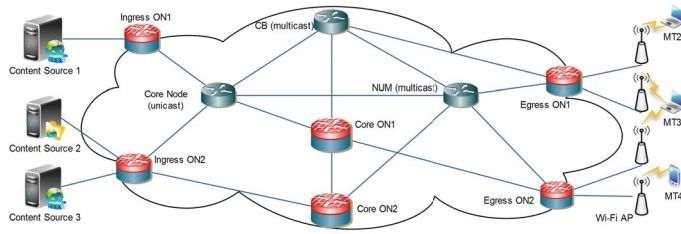


Figure 3. General Evaluated Scenario

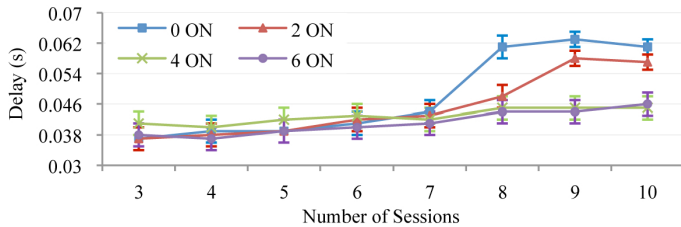


Figure 4. Data Delay Time

some specific parameters according to the evaluation test characteristics. The links in the core network were configured with a random delay between 1 and 2 ms and a bandwidth ranging between 5 and 7 Mbps. Although the nodes are fixed, the links between the nodes are randomly generated. The implementation revealed several limitations of the NS-2 regarding the incompatibility between multicast (flat routing), domains (hierarchical addressing) and scenarios involving wireless and fixed parts. To overcome these issues, we decided to emulate the wireless connections on wired links. Thus, we employ a dynamic link error model, adapting delay and losses, according to previous simulations made on wireless scenarios. Moreover, the distances between the APs were defined so that their coverage area do not overlap and the communications do not interfere between each other. Two different traffic generators were randomly used: exponential traffic and constant bit rate. All flows have a packet size of 1000 bytes and an average rate of 100 Kbps. Regarding the QoS implemented mechanisms, 6 CoS were considered: signaling, routing, Expedited Forwarding, Best Effort (BE), Assured Forwarding 1 (AF1) and AF2. The CoS of each flow is randomly chosen, but it is guaranteed that more than a half of the traffic is BE to simulate a real network.

#### A. Influence of ONs

As described in Section III-E, the number of ONs in the core can be chosen according to the operator preferences, becoming relevant to study different cases of the ONs' influence, especially in delay of the data packets, network reconfigurations, session establishment time and session reestablishment time.

1) *Delay of Data Packets:* From the results obtained in the Fig. 4, one may observe that packet delay decreases with the number of ONs introduced in the core network. Since the number of ONs is proportional to the number of sub-AMTs, as more sub-AMTs exist in the core, less reservations and/or releases are needed to deliver content to the MTs. These results are more notorious for a higher number of sessions, since it is likely that some sessions would use flows that already

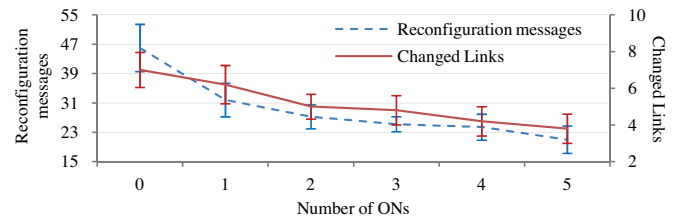


Figure 5. Impact of ONs' number in network re-arrangements

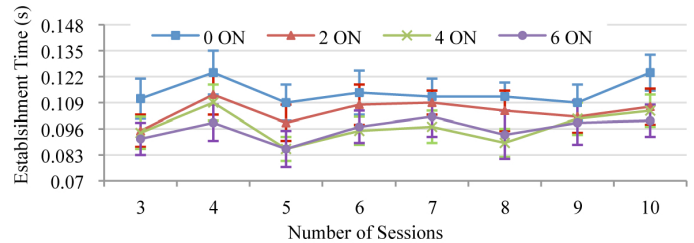


Figure 6. Session Establishment Time

have established paths in the core network, only requiring an extension of the multicast tree. This way, the network reacts quickly to changes and in a scalable way, leading to lower delays and losses. It should be stressed that, after a certain point, increasing the number of ONs does not correspond to substantial improvements and may not justify collateral costs, such as overhead.

2) *Network Reconfigurations:* As scalability and resilience are the main concerns of the proposed architecture, it becomes important to assess the behavior of the implemented components regarding these aspects. To perform this evaluation, we considered several scenarios with different number of ONs where a MT performs an handover between several APs. The number of reconfiguration messages and changed links involved in the process are presented in Fig. 5.

Observing the obtained results, one may conclude that the introduction of ONs in the core network increases the architecture scalability, since both metrics decrease as the number of ONs increase. We observe that core ONs facilitate the reconfiguration process of the QoS-aware multicast tree, increasing the scalability of the proposed architecture and the original AMTs concept. The scenario with 5 ONs, representing a core almost full of ONs, shows no benefits, since reconfiguration messages and changed links continue similar while the overhead increase. A balanced number of ONs and simple nodes in the core network is the most scalable solution.

3) *Session Establishment Time:* It is important to study the impact of ONs in the session establishment time, to evaluate the architecture response to the MTs movements. This metric (Fig. 6) tends to be smaller with the increase of the number of ONs in the core network. More ONs in the core increases the probability of reusing existing sub-AMTs for different users. So, the number of ONs in the core network should be carefully chosen considering always a compromise between the establish time and the control overhead.

4) *Session Reestablishment Time:* Session reestablishment time results (Fig. 7) show better performance in the case of having a scenario with 4 ONs in the core network, about

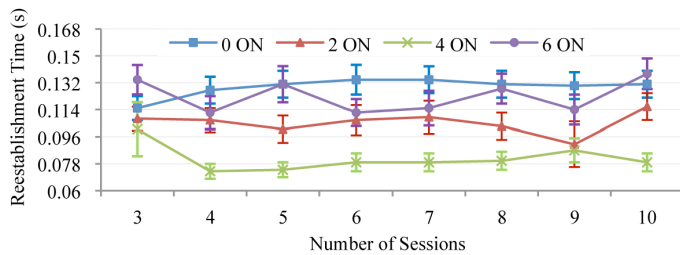


Figure 7. Session Reestablishment Time

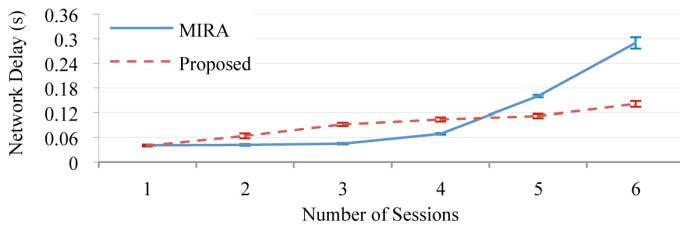


Figure 8. Network Data Delay

half ONs of the core network. The studied scenario was not overloaded, especially concerning the APs, which means that session reestablishment time barely depends on the load of the used AP. When a message notifying MT movement is sent through an overload AP or when MT receives a message back also by an overload AP, the session reestablishment time increases significantly. Moreover, for scenarios with a high number of core nodes, the time of traveling across the core becomes more relevant for the session reestablishment time.

5) *Influence of Unicast Nodes:* The architecture supports an integration of multicast and unicast nodes through the sub-AMTs layer. Thus, despite not shown due to space limitations, it is important to evaluate the impact of core nodes that do not support multicast. We observed that the overhead of control becomes slightly higher with the increase of unicast nodes and the delay increases with the number of sessions independently of the number of unicast core nodes. Hence, one may conclude that the presence of unicast nodes in the core network does not affect the performance of the network.

### B. Comparing with MIRA

Since IPT functionalities are based on MIRA [9], here we compare the developed solution with the initial MIRA solution. In MIRA, the MT does not have any information about the network and randomly chooses one available AP to receive the pretended data flow. MIRA chooses the core path from data source to AP, based on unicast routing table, and it does not allocate resources in wireless environment. According to Fig. 8, for small number of sessions, MIRA gains a small advantage, since the proposed solution tends to aggregate the flows from same CoS in the same AP. However, for values larger than 5 sessions, MIRA presents higher delay values comparing to the proposed solution. This is justified by the fact that the proposed solution uses several core paths and APs to distribute flows according to network resources, CoS and user characteristics. Moreover, our approach contains all

other features not present in MIRA, such as the hierarchical scalability support and the context-aware control.

## V. CONCLUSIONS AND FUTURE WORK

Considering the strict requirements of future group-based multimedia applications, this paper proposed an architecture to support a context-aware transport of multiparty content, enabled by an hierarchical control framework. Several components and functionalities were developed to achieve a scalable and seamless generic transport while providing end-to-end guaranteed QoS. The outcomes of the evaluation show that these mechanisms are able to provide a more scalable control solution through the AMT concept, avoiding global reconfigurations and performing local rearrangements. In general, one may conclude that this hierarchical approach is an improvement towards overcoming demands of future internet services. As future work, we plan to evolve the intelligence introduced in the network management, by developing self-management and autonomic concepts to achieve self-organized architectures, to increase the network resilience capable of dealing with the NGN complexity.

## REFERENCES

- [1] C. Diot et al., "Deployment issues for the ip multicast service and architecture," *Network, IEEE*, vol. 14, no. 1, pp. 78–88, Jan/Feb 2000.
- [2] R. Boivie et al., "Explicit multicast (xcast) basic specification," in *IETF Internet Draft*, 2000.
- [3] L. H. M. K. Costa, S. Fdida, and O. Duarte, "Hop by hop multicast routing protocol," in *ACM SIGCOMM*, 2001, pp. 249–259.
- [4] M. Hosseini, D. Ahmed, S. Shirmohammadi, and N. Georganas, "A survey of application-layer multicast protocols," *Communications Surveys and Tutorials, IEEE*, vol. 9, no. 3, pp. 58–74, Quarter 2007.
- [5] R. Bless and K. Wehrle, "Ip multicast in differentiated services (ds) networks," *IETF RFC 3754*, April 2004.
- [6] S. Blake et al., "Rfc2475: An architecture for differentiated service," *RFC Editor*, 1998.
- [7] E. Cerqueira, L. Veloso, A. Neto, M. Curado, E. Monteiro, and P. Mendes, "A unifying architecture for publish-subscribe services in the next generation ip networks," in *IEEE Globecom*, 2006.
- [8] E. Cerqueira, L. Veloso, A. Neto, M. Curado, P. Mendes, and E. Monteiro, "Q3m - qos architecture for multi-user mobile multimedia sessions in 4g systems," *Real-Time Mobile Multimedia Services*, vol. 4787, pp. 38–49, September 2007.
- [9] A. Neto, E. Cerqueira, A. Rissato, E. Monteiro, and P. Mendes, "A resource reservation protocol supporting qos-aware multicast trees for next generation networks," in *ISCC*, July 2007, pp. 707–714.
- [10] F. Mitrano et al., "Qos management of multicast and broadcast services in next generation networks," in *Mobile and Wireless Communications Summit, 2007. 16th IST*, 2007.
- [11] K. Li and J. Tian, "A qos mobile multicast routing algorithm based ant colony algorithm," *Pacific-Asia Workshop on Computational Intelligence and Industrial Application, IEEE*, vol. 2, pp. 88–92, 2008.
- [12] A. K. Dey, "Understanding and using context," *Personal and Ubiquitous Computing*, vol. 5, pp. 4–7, 2001.
- [13] O. Riva et al., "A next generation operator environment to turn context-aware services into a commercial reality," in *MDM*, 2008.
- [14] M. Williams et al., "Context-awareness and personalisation in the daidalos pervasive environment," in *ICPS*, July 2005, pp. 98–107.
- [15] B. Mathieu et al., "Self-management of context-aware overlay ambient networks," in *IFIP/IEEE IM*, 21 2007-Yearly 25 2007, pp. 749–752.
- [16] N. Coutinho, T. Condeixa, R. Valbom, S. Sargento, and A. Neto, "Context-aware selection in multicast environments," *Accepted in IEEE Symposium on Computers and Communications*, 2010.
- [17] S. Bhattacharyya, "An overview of source-specific multicast (ssm)," *IETF RFC 3569*, July 2003.
- [18] "The Network Simulator NS-2," <http://www.isi.edu/nsnam/ns/>.