Scalable Multimedia Group Communications through the Over-provisioning of Network Resources

Augusto Neto[1], Eduardo Cerqueira[2], Marília Curado[2], Paulo Mendes[3] and Edmundo Monteiro[2]

[1] Institute of Telecommunications, 3810-193 Aveiro, Portugal
augusto@av.it.pt
[2] University of Coimbra, 3030-290 Coimbra, Portugal
{ecoelho, marilia, edmundo}@dei.uc.pt
[3] INESC Porto, 4200-465, Porto, Portugal
pmendes@inescporto.pt

Abstract. The efficient management of network resources together with the *Quality of Service* (QoS) control of real-time multimedia group communication sessions in *Next Generation Networks* (NGN) is still a challenging research goal. The unified control of the session quality level, distribution tree allocation and network resources in NGN will increase the user satisfaction, reduce operational costs, optimize network resources and maximize the profits of providers. This paper introduces the *Multi-user Aggregated Resource Allocation* mechanism (MARA), which supports a dynamic control of surplus class-based bandwidth and multicast resources in a scalable way, while assuring the minimal quality level of multimedia group communication sessions. In comparison with existing works, MARA significantly reduces signalling, state and processing overhead. In addition, simulation results present the benefits of MARA by improving the network performance under re-routing conditions.

Keywords: Real-time multimedia sessions; Next generation networks; Network resource provisioning; Multi-user communications.

1 Introduction

Real-time multimedia sessions are now present in our daily live experience, and will be among the most important applications in next generation networks. The efficient distribution and the quality level control of multimedia sessions, such as IPTV, video streaming and other multimedia multicast-alike sessions, will attract new clients, while increasing revenues to providers. In such real-time session, the content distribution is performed simultaneously to groups of users (multiple users, called multi-user sessions in this paper). In this context, IP multicast seems to be the most

This work was done at the Laboratory of Communications and Telematics of the Faculty of Science and Technology of the University of Coimbra. It is supported by DoCoMo Euro-labs, by the Portuguese Ministry of Science, Technology and High Education, and by European Union FEDER - POSI (projects Q3M and SAPRA).

attractive solution to control the distribution of multi-user sessions, because it avoids packet duplication and saves network resources. In addition, QoS-aware multi-user sessions can be controlled by using the *Differentiated Service* (DiffServ) model in order to provide scalable traffic differentiation, QoS assurances and overcome the limitations of the current IP best effort approach.

Our previous work proposed the *MultI-Service Resource Allocation* (MIRA) [1] mechanism to allow the distribution of multi-user sessions with QoS support in *Next Generation Networks* (NGN). MIRA supports the coordinated control of *Source-Specific Multicast* (SSM) distribution trees and QoS assurances for multi-user sessions in DiffServ classes. The setup of network resources is deployed by the *Resource Control Function* (RCF) component, through the support of a per-flow edge-to-edge ingress-driven single-pass signalling approach implemented by the *MIRA Protocol* (MIRA-P). MIRA-P defines message-specific flags to control the RCF behaviour in the nodes throughout the network, and uses only two messages for resource requests and operation feedback control, *RESERVE* and *RESPONSE* respectively. The RCF interacts with network elements (e.g., QoS schedulers) for resource allocation, release and update procedures, as well as to detect re-routing events. MIRA state is controlled by soft-state to increase the system robustness. The performance evaluation described in [1] shows that MIRA distinguishes itself from existing solutions by supporting low complex operations, and by controlling multi-user sessions with minimized signalling and state overhead.

However, the per-flow signalling used by MIRA to control the bandwidth required for multi-user sessions in DiffServ classes and the allocation of SSM trees associated with them, poses important issues regarding system performance. While the bandwidth control requires excessive signalling and processing costs in large-scale networks, the SSM trees management (per-flow control) suffers from state overhead. Moreover, unpredictable topology changing detections (e.g., link failures, handovers or new *join/leave* requests) need MIRA operations to restore each affected flow of a multi-user session (scalable session) on a new path that at least fulfils their minimum QoS requirements. Our performance findings in the analysis of MIRA, associated with the strong resource requirements of multimedia group communications, motivated further investigation of strategies to avoid per-flow operations.

In order to overcome the current MIRA limitations, increase the satisfaction of users and optimize resources in multi-user systems, this paper introduces the *Multi-user Aggregated Resource Allocation* (MARA) proposal. MARA provides scalable support of IP multi-user sessions, while minimizing signalling, state and processing costs of MIRA per-flow operations. The MARA contribution is performed by dynamically controlling the over-provisioning of DiffServ classes and SSM trees. Thus, MARA establishes multiple multi-user sessions without per-flow signalling by combining admission control and surplus network resources (bandwidth and SSM trees) procedures in advance. Multicast aggregation is used to optimize multicast state storage. This paper demonstrates the benefits of MARA in comparison with MIRA to control multi-user sessions regarding session robustness operations under re-routing conditions. The MARA performance evaluation was carried out by simulations, which analyzed bandwidth/multicast state and signalling overhead under re-routing events.

The remainder of this paper is organized as follows. Related work is presented in section 2. Section 3 provides a detailed description of MARA, whose performance is evaluated in Section 4. Finally, Section 5 concludes this paper with a summary of our findings.

2 Related Work

Over-reservation strategies to over-provision bandwidth resources have been addressed in the literature on a set of proposals. The *Border Gateway Reservation Protocol* (BGRP) [2] and the *Shared-segment Inter-domain Control Aggregation* protocol (SICAP) [3], use a similar two-pass receiver-driven signalling approach to aggregate unicast traffic into sink- and shared trees respectively. However, the multicast trees control is supported by neither of the two approaches. Nevertheless, SICAP demonstrates that the dynamic control of surplus bandwidth reservations is more efficient than the static quantification factor used by BGRP. Furthermore, the *Dynamic Aggregation of Reservations for Internet Services* (DARIS) [4] proposal uses a centralized approach to control resource allocations. DARIS deploys over-reservations based on information about the internal topology, resource capacities and current selected routes. Thus, scalability is the main issue in DARIS proposal, since too much centralization endangers system performance to manage multimedia sessions. For instance, excessive signalling can be required to control resource synchronization and setup.

The creation of multicast trees in advance has been used by some proposals as a means for the over-provisioning of multicast resources. However, there is a lack of solutions focused on allowing a scalable multicast over-provisioning support. Most of the available proposals aim to protect the system against topology changes (e.g., link breaks), instead of reducing signalling overhead, and thus processing overhead, to setup and maintain new multi-user sessions. The solutions presented in [5] and [6] aims to create a pair of trees (a main and a backup tree) for each session request, where the switching procedure from the main tree to the backup tree is controlled dynamically. Thus, the system must be signalled twice whenever a new session request is received, in addition to requiring that each visited router store twice amount of state. In spite of speeding up the restoration of sessions affected by re-routing, scalability (in terms of state and signalling overhead) is a shortcoming in backup-tree based solutions within large-scale systems.

From the related work analysis, none of the above proposals coordinates the over-provisioning of QoS resources/bandwidth and multicast resources. The over-reservation approaches are neither scalable (two-pass signalling protocol) nor efficient for multi-user sessions (no multicast support). In addition, backup-tree based proposals introduce signalling, processing and state overhead to improve reliability. Thus, MARA is proposed to address the above challenges.

3 Multi-user Aggregated Resource Allocation

MARA dynamically provides the over-provisioning of network resources (bandwidth and multicast) to establish group communication sessions without per-flow signalling.

QoS-aware sessions are controlled based on DiffServ per-class over-reservations, where resources are dynamically re-adjusted according to the bandwidth demand. SSM trees are over-provisioned in advance (at the system bootstrap) and dynamically controlled based on multicast aggregation. By default, MARA detects re-routing via soft-state operations. In order to improve system robustness, MARA interacts with unicast routing protocols to acquire information also about re-routing events. MARA invokes the restoration of sessions affected by re-routing upon intercepting router advertisements generated as a consequence of topology changes (e.g., OSPF *router Link State Advertisements (LSA)* messages). With MARA, the system is optimized and the satisfaction of the users is increased, because it minimizes the state, signalling and processing overhead of MIRA as well as it reduces the convergence time to setup QoS-aware multi-user sessions on new paths.

MARA re-uses some MIRA components, namely the signalling protocol (MIRA-P) and the resource controller (RCF). MIRA-P provides control information so that RCF can configure the state of network elements to setup network resources associated with multi-user sessions. In relation with MIRA, the common header of MIRA-P messages was extended with two new message-specific flags. Thus, MARA can invoke the initialization of network resources (flag *Initialization* (I)), setup of SSM trees (flag *Multicast* (M)), and the dynamic adjustment of over-reservations according to current demands (flag *Over-reservation* (O)). Moreover, two components ware developed to control the over-provisioning of bandwidth and SSM trees, namely *Advanced QoS Resource Allocator* (ASAC) and *Advanced Aggregation Tree Allocator* (AGTree), respectively. Fig. 1 shows MARA components implemented at the network edges. Interior nodes only implement MIRA-P and RCF. The MARA intelligence is pushed to the edges elements in order to increase the system scalability and robustness.

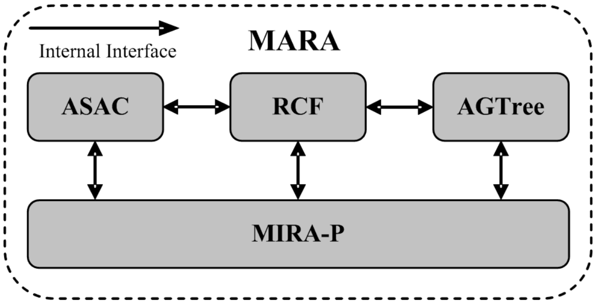

Fig. 1. MARA Architecture

ASAC over-provisions bandwidth by assigning per-class over-reservations at the system bootstrap and controlling their dynamic adjustment in an on-demand basis, without generating per-flow signalling. In order to avoid class starvations, each *Class of Service* (CoS) is assigned a *Committed* and a *Maximum Reservation Threshold* (*CRth* and *MRth*, respectively). Moreover, a global initialization factor is also assigned as a fraction of the local link capacity (e.g., ½ or ¼). Whenever realized that a CoS cannot accommodate a multi-user session request (i.e., an overload class), ASAC tries to increase the current CoS's over-reservation based on network utilization ratios and commitments. Moreover, the size of the CoS's *MRth* is also controlled, so that a session can be admitted when the *MRth* of its required CoS is currently congested (procedure not allowed in MIRA).

AGTree over-provisions SSM distribution trees by assigning surplus trees at the system bootstrap. AGTree uses encapsulation to merge multi-user sessions into aggregation trees (instead of source-routing proposals) without any signalling exchange inside the network. Thus, data packets are aggregated at ingress routers and de-aggregated at egress routers of a network accordingly. Moreover, AGTree controls the connectivity of multi-user sessions by dynamically switching sessions from a SSM aggregation tree to another as a consequence of network dynamics (e.g., new *join/leave* requests or re-routing).

3.1 MARA Functionalities

This section introduces the functionalities supported by MARA to control the over-provisioning of network resources. It is assumed that all MARA agents have information about whether the local router is an ingress, core or egress router.

3.1.1 System Initialization

Ingress routers use a flooding mechanism to request the initialization of bandwidth and to collect information about the available shortest paths until all available edges router of a network. In contrast to end-to-end, the edge-to-edge approach of MARA allows autonomy of each network to control and coordinate resource allocations. Each router within the network initializes the bandwidth resources through multiplying the index factor by the *MRth* of each local CoS. The information about the edge-to-edge paths is used to setup the SSM aggregation trees. An overview of the operations deployed by MARA to initialize network resources is described in the remainder of this Section.

Ingress routers setup the over-reservation of each CoS on all local network interfaces based on the *MRths* and the initialization factor. Afterwards, a *RESERVE(I)* (*RESERVE* message with flag Initialization (*I)*) is composed. The *RESERVE(I)* must carry the initialization factor and the per-class *MRth*, as well as be prepared to collect information about the composition of the resultant distribution path and the CoS's characteristics of the bottleneck link. Hence, a copy of the *RESERVE(I)* is sent in all interior routers (except the inter-network link), where each one has the IP address of the associated local *outgoing interface* (Oif) filled in the *Reserved Path* (RSVPATH) object. The current QoS capabilities are filled in the *QoS Specifications* object (QSPEC) as proposed in the NSIS suite. As a consequence of the flooding schema, each router must do some verification before setting up resources aiming at: 1) ensuring that per-class over-reservation is initialized only once (when the CoS has no reservation state); 2) avoiding infinite signalling loops (by dropping messages already carrying the IP address of a local network interface in the *RSVPATH* object). As a result of 2), long trees are avoided by dropping *RESERVE(I)* messages that return to ingress routers. The signalling used by MARA, and its composition, to initialize the bandwidth resources in a generic scenario is shown in Fig. 2.

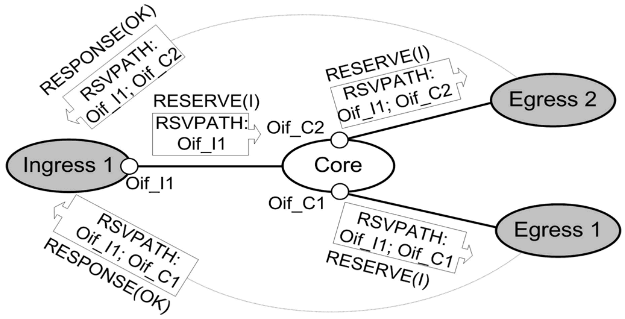

Fig. 2. Signalling events for the initialization of bandwidth resources in each network router

After initializing the bandwidth resources, ingress node *I1* composes a *RESERVE(I)* carrying the initialization factor, the *MRths* and the *RSVPATH* object (filled with the *Oig_I1* IP address), which is then sent downstream. Upon visited by the *RESERVE(I)*, core node *C1* initializes the local bandwidth resources (as in *I1*) and sends a copy of the *RESERVE(I)* (with the *RESVPATH* object of each one correctly updated) to all local network interfaces (*Oif_C1* and *Oif_C2* in this case), except the one in which it was received. In the case of an edge router (egress nodes *E1* and *E2* in Fig. 2), a *RESPONSE(OK)* message is composed with the information derived from the associated *RESERVE(I)* after concluded the local bandwidth initialization. The *RESPONSE(OK)* is sent to the IP address found in the first entry of the *RSVPATH* object (i.e., *Oif_I1*), allowing the ingress router to store information about the edge-to-edge path and about the characteristics (per-class bandwidth and thresholds) of the CoS of the bottleneck link of all paths.

After successfully accomplished the initialization of CoS bandwidth, MARA setups the SSM trees along each collected communication path. As a first step, a SSM tree is assigned to each communication path composed by a pair of ingress and egress routers (called un-branched). Each tree is composed/identified by the IP address of the ingress router (*I1* in the case) and a multicast group IP. The latter is locally created by *I1* via a dynamic multicast address allocation solution [7]. After that, a *RESERVE(M)* (notation used to a *RESERVE* message with flag Multicast (*M)*) is sent downstream, being forwarded based on the *RSVPATH*. In order to allow the support of QoS-aware multicast trees under environments with asymmetric routing, each visited router configures the *Multicast Routing Information Base* (MRIB) with information derived from the *RSVPATH* object and propagates the message downstream. At the egress routers *E1* and *E2*, PIM-SSM is triggered to build the SSM tree, and a *RESPONSE(OK)* message is sent to *I1*, confirming the successful operation.

After the creation of the un-branched trees, MARA uses a combinatorial algorithm to create a set of branched trees, simultaneously supplying multiple egress routers. At first, all possible combinations between the available un-branched trees are generated in n-1 interaction, where n is the number of un-branched trees. In order to optimize the large number of resultant branched trees, filters are used to retain the best ones. One selection criterion is based on the fact that a distribution tree demands more network resources (e.g., bandwidth) as branching points get close to the root. Based on this criterion, SSM trees with branching point in the ingress routers are discarded. Furthermore, combinations with multiple paths converging to the same node are also discarded. Optionally, MARA can be configured with a maximum on-tree hops (which can derive from network historical or measurement tools). After generating all SSM aggregation trees, the ingress routers signal the identified egress routers so that PIM-SSM can build the created SSM trees.

The combinatorial algorithm used to create the branched SSM aggregation trees can pose performance overhead to MARA, where the processing cost is overloaded with the increasing number of combinations (which increases with the number of links/nodes in the network). Other combinatorial algorithms can be used. The specification of an efficient combinatorial solution is out of the scope of this paper.

3.1.2 Dynamic Resource Allocation

The combination of over-provisioning and admission control allows MARA to dynamically establish multi-user sessions without per-flow signalling. Whenever required the establishment of a session, MARA must be triggered at an ingress router with a session request (*Ri*) carrying the QoS requirements within the QSPEC (at least the CoS, the required bandwidth (*Brq(i)*) and user IP). Based on the egress router IP associated to the interested user (retrieved via intra-domain routing approaches, such as BGP), MARA selects one (the first matching) of the previously created SSM aggregation trees. Afterwards, the packets association with multi-user sessions are encapsulated (aggregation) in the selected SSM aggregation tree. At the egress router(s) associated to the selected SSM aggregation tree, the original IP header information is restored (de-aggregation) and forwarded towards its destination.

If succeeding the information that the ingress router has about the current usage of the required CoS *i* in the bottleneck link of the required SSM tree (i.e., the bandwidth required by *Ri* is available in the current over-reservation of CoS *i*), the session is admitted without further processing or signalling. In addition, whenever the required CoS *i* experiences unavailable bandwidth, ASAC attempts to re-adjust its current over-reservation. The amount of bandwidth to re-adjust the over-reservation of CoS *i* (*Bov(i)*) is given by (1). The Equation 1 is based on the utilization ratio (*Bu(i))* of the CoS and the amount of bandwidth which will be available after the session setup. The information used in (1) is related to the bottleneck link of the selected SSM aggregation tree. After the computation of a positive *Bov(i)* (in case of failure the equation returns a negative value), MARA updates the current reservation of CoS *i, Brv(i)*, where *Brv(i)* ← *Bov(i)* + *Brq(i)),* and sends a *RESERVE(O)* (notation used to a *RESERVE* message with flag Over-reservation (*O)*) in the selected communication path.

$$Bov(i) = \underbrace{\frac{Bu(i)}{MRth(i)}}_{\text{Utilization}} + \underbrace{(MRth(i) - Bu(i) - Brq(i))}_{\text{Bandwidth available after setting up } Ri} \tag{1}$$

All routers along the communication path update the local *Brv(i)* with the *Brq(i)* derived from the QSPEC transported in the *RESERVE(O)* message. Upon receiving the *RESPONSE(OK)*, the ingress router re-processes *Ri*, which is now supposed to succeed (available bandwidth) and be admitted without signalling. If the amount of resources requested by *Ri* exceeds the current *MRth*(*i*), (1) does not succeed (*Bov(i)* <

0), and so the adjustment of the CoS over-reservation fails. In this congestion case, MARA invokes the re-adjustment of the CoSs so that *Ri* can be admitted. For this propose, (2) provides a bandwidth index (*B_Idx(j)*) of any other CoS *j*, except the congested one (*i*). The bandwidth index of a CoS is the ratio between the amount of bandwidth that is currently available and reserved by CoS *j*.

$$\forall j \in c, B\_Idx(j) = \left(\frac{Brv(j) - Bu(j)}{Brv(j)} \right) \quad (2)$$

Furthermore, the threshold index (*Th_Idx(j)*) of any CoS *j*, except the congested one (*i*), is provided by (3) as being the ratio between the *MRth* and a bandwidth reference (*Bref(j)*). The bandwidth reference can be the bandwidth currently reserved for CoS *j* (if the *Bu(i)* is lesser than the *CRth(i)*), otherwise the *CRth* is used. Thus, MARA ensures at least the minimal QoS commitment of the CoSs.

$$\forall j \in c, Th\_Idx(j) = \left(\frac{MRth(j) - Bref(j)}{MRth(j)} \right) \quad (3)$$

The results of (2) and (3) are used to compute, by means of (4), the amount of bandwidth by which the *MRth* of each CoS *j* (*Brl_MRth(j)*) will be reduced. Equation 4 computes *Brl_MRth(j)* based on the average between the sum of bandwidth and threshold indexes of CoS *j*, multiplied by its amount of bandwidth currently available.

$$\forall j \in c, Brl\_MRth(j) = \left(\frac{B\_Idx(j) + Th\_Idx(j)}{2} \right) * (MRth(j) - Bref(j)) \quad (4)$$

After computation of (4), MARA adds *MRth(i)* with the sum of the *Brl_MRth(j)* of the selected CoS (*j*). The *MRth(j)* of each CoS (*j*) is decreased by the computed value accordingly. After successfully accomplished the CoS re-adjustment, *Ri* is re-processed again and the over-reservation is expected to be successfully updated now. If the re-adjustment of the CoSs fails *Ri* is rejected. As occurs with MIRA, MARA also allows explicit releasing of resources, which is done by signalling the network with a *RESERVE(T)* (*RESERVE* message with flag *Tear* (T)).

3.1.3 Session Connectivity Control

MARA ensures the continuity of on-going multi-user sessions during their entire lifetime by automatically switching them between available SSM aggregation trees. This operation is deployed at the ingress router associated with the required session. This way, MARA prevents waste of resources (by avoiding sending packets to leaf nodes without active member users) and session quality degradation is controlled due to re-routing (e.g., link failures, new *join/leave* events or handovers).

In the case of receiving new *join/leave* events (or handovers) and detecting that a session has no leaf nodes in some of the egress routers of its current SSM aggregation tree, MARA performs the following operations: (*i*) selects another SSM aggregation tree leading to the right set of egress routers and supporting the required QoS (as upon receiving the *Ri*); (*ii*) switches the session to the selected SSM tree (by controlling the aggregation accordingly). In the case that none of the available SSM aggregation trees can support the QoS required by the affected session, MARA tries to re-adjust the current resource configuration. If the switching fails (unavailable resources), the request is denied.

In what concerns re-routing detections due to link-failures, the same set of operations described above is deployed by MARA. Based on the local *router LSA,* generated by OSPF upon detecting that the state of a network interface changes (*down* or *up*), a copy of a *RESPONSE(Failure)* (*RESPONSE* message with a failure report) must be sent to each ingress router (locally stored in the reservation data base). After received the *RESPONSE(Failure)*, MARA locally matches (session state) all flows established in the SSM aggregation tree(s) affected by the failure condition to further re-route them into another one(s). The sessions can be restored in the previous path when the link is *up* again, if the new path cannot fulfil all requirements.

4 MARA performance evaluation

The benefits of MARA over MIRA, to setup multi-user sessions in a dynamic environment, from the network and the user expectation, are verified by using the *Network Simulator-2.29* (NS-2.29). The simulation model considers a network topology composed by 14 routers, interconnected by links with different capacities and propagation delay randomly generated by BRITE. As suggested in [8], the simulation model supports one *Expedited Forwarding* alike CoS (Premium), two *Assured Forwarding* alike CoSs (Gold and Silver), and one Best Effort class. The initialization factor and the *MRth* are respectively 25% and 20%, and the *CRth* is 50% of the *MRth* for each one of the four CoSs. In order to achieve all functionalities required to accomplish the evaluation, the NS-2.29 was extended with the WFQ discipline (for QoS scheduling), PIM-SSM (for IP multicast), as well as MIRA and MARA agents accordingly.

The experiments were repeated 10 times to simulate a large number of multi-user UDP sessions (1,000) sent from the same ingress router to different receivers, with confidence interval of 95%. The sessions have a lifetime that varies from 20s (short-live) to 120s (long-live), and have a constant bit rate of 224Kb/s to emulate a scalable session with three flows are used by common multimedia CODECs, such as MPEG4 (flow 1 32Kb/s, flow 2 64Kb/s and flow 3 128Kb/s) [9]. The session requests (establishing and releasing) take place from the beginning up to the end of the simulation (120s), being generated by a Poisson distribution. Furthermore, MARA is configured to over-provision SSM trees with a maximum limited number of 6 hops, as justified in the following. Our previous study [10] revealed that only 5.4% of all available trees, selected in average, comprise a maximum of 5 hops. Moreover, other researches [11], [12], attested that 80% of intra-network shortest paths have 4 hops or less.

The performance evaluation is composed by three experiments, which aims to examine the impact that MIRA and MARA takes to establish multi-user sessions, and re-route them, from the perspective of both network and user. Hence, re-routing conditions are introduced by setting the state of a potential link to *down* and *up* at random instants. Regarding network perspective, it is examined the signalling load throughout the simulation and, in the users perspective, it is studied the throughput in the receivers. In the first set of tests, the QoS support is provided by DiffServ without the presence of a solution to implement admission control (without MIRA or MARA). In the next set of tests, whereas DiffServ provides QoS support, MIRA and MARA implement resource allocation and admission control.

Additionally, background traffic sources are placed in the environment in order to achieve data losses by exceeding in 20% the overall link capacity of the new communication path. For simplification, only the results measured for one receiver that has joined a multi-user session (denoted as *R1* and *M1* respectively) is considered due to the similar results of all the remaining receivers. The simulation results expose that *M1* starts at instant 5.84s, and the events for link-down and link-up take place at instant 33.75s and 74.8s respectively. Moreover, OSPF averages 5.88s to generate a router LSA after detected the link-down event, and 5.11s after detected the link-up event.

The numeric results reveal that MARA minimizes in 66.35% the signalling load taken in MIRA tests. The per-flow basis of MIRA is the reason for the excessive signalling load, where the twice re-routing events (when the network interface goes down and come up again) has triggered per-flow signalling to restore all affected multi-user sessions. Thus, MIRA exchanged approximatelly11.10 times more data for *RESERVE* messages than MARA.

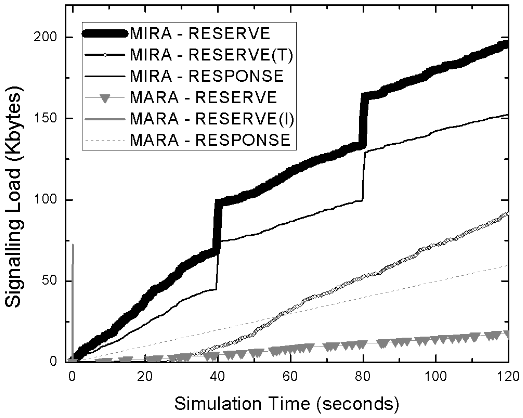

Fig. 3. Signalling load of MIRA and MARA tests

The traced results show a signalling load averaging 448.15KB (MIRA) and 150.76KB (MARA). The per-flow resource allocation signalling of MIRA is the reason for its large signalling load. In contrast, MARA signalling occurs mostly during the system initialization phase (73.42KB for *RESERVE(I)*), being the signalling load during the simulation resultant of the dynamic update of over-reservations (17.64KB for MARA, while MIRA generates 195.89KB). As a consequence, MIRA also generates more *RESPONSE* messages (102.80KB) than MARA (59.85KB). Furthermore, per-flow releasing operations (triggered by *RESERVE(T)* messages, and invoked whenever a multi-user session ends) introduce an average of 91.51KB in MIRA experiments. MARA has no *RESERVE(T)* messages, since the end of multi-user sessions grants the associated CoS with surplus resources, instead of releasing them (as occurs with MIRA). Hence, MARA minimizes in 66.35% the signalling load of MIRA.

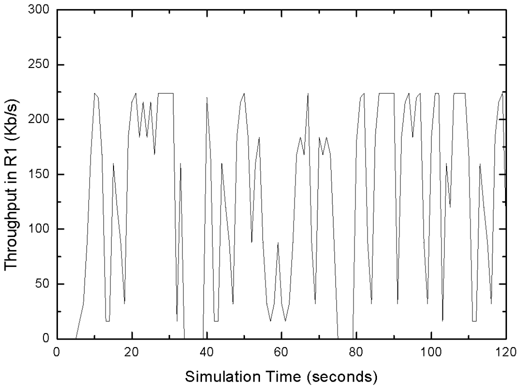

Fig. 4. Throughput at mobile user *R1* with DiffServ

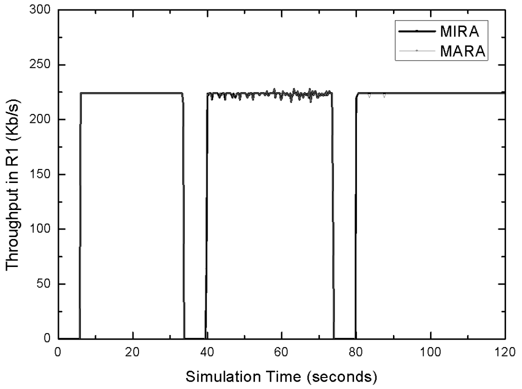

Fig. 5. Throughput at mobile user *R1* with MIRA and MARA

It can be clearly noted the degradation experienced by *M1* in the DiffServ tests, in which only during ~24% of the simulation its content is propagated with full rate. In the new path (from ~39 seconds up to ~75 seconds), the degradation of *M1* is even worse due to the congestion caused by background traffics. In the MIRA and MARA tests, a lightly variation in the throughput is noted in the new path due to the presence of congestion. However, this variation does not endanger the quality of *M1*, since no packet loss is noticed. Thus, MIRA and MARA are perfectly suitable to support multimedia content to groups of users.

6 Conclusions

This paper introduces the *Multi-user Aggregated Resource Allocation* mechanism (MARA) to coordinate the over-provisioning of bandwidth and SSM trees to be used by multimedia multi-user sessions. The proposed mechanism aims to mitigate the scalability problems of per-flow signalling approaches, such as the *MultI-service Resource Allocation* mechanism (MIRA) previously proposed by the authors. The combination of over-provisioning of resources (bandwidth of DiffServ classes and SSM trees) with admission control allows MARA to setup a significant number of sessions (74.9%) without any signalling in the network. Moreover, the performance limitations of IP multicast (state and maintenance signalling) are overcome by using multicast aggregation. Finally, waste of resources due to over-reservations is prevented by dynamically re-adjusting resources and controlling session connectivity on-demand. Simulation results prove the benefits of MARA in relation to MIRA. Although MARA performs all over-reservations in-advance, after the system bootstrap, simulation results show that system initialization takes only 0.04s and consumes no more than 8% of the link capacity. During the simulation, MARA reduces in 60.33% the signalling load in relation to MIRA. In what concerns state overhead, MARA reduces the state of the ingress router (the most problematic point) in 67.5% in comparison to MIRA, and generate signalling to control SSM trees no longer. The future work prompted in this paper consists in providing efforts to evaluate MARA through prototyping. Thus, it is expected to examine the accuracy of the benefits of MARA described in this paper in real scenarios.

References

1. Neto, A., Cerqueira, E., Rissato, A., Monteiro, E., Mendes, P.: A Resource Reservation Protocol Supporting QoS-aware Multicast Trees for Next Generation Networks. In: 12th IEEE Symposium on Computers and Communications, Aveiro, Portugal (2007)
2. Pan, P., Hahne, E., and Schulzrinne, H.: BGRP: A Tree-Based Aggregation Protocol for Inter-domain Reservations. J. Com. and Net. 2, 157--167 (2000)
3. Sofia, R., Guerin, R., Veiga, P.: SICAP, a Shared-segment Inter-domain Control Aggregation Protocol. In: Conf. in High Performance Switching and Routing, Turin, Italy (2003)
4. Bless, R.: Dynamic Aggregation of Reservations for Internet Services. In: 10th Conference on Telecommunication, pp. 26--38, Monterey (2002)
5. Braun, T., Arya, V., Turletti, T.: A Backup Tree Algorithm for Multicast Overlay Networks. In: Networking, Waterloo (2005)
6. Kodialem, M., Lakshman, T.: Dynamic routing of bandwidth guaranteed multicasts with failure backup. In: *IEEE INFOCOM*, New York (2002)
7. Hanna, S., Patel, B, Shah, M.: Multicast Address Dynamic Client Allocation Protocol (MADCAP). IETF RFC 2730 (1999)
8. Di, Z., Mouftah, H.: Performance Evaluation of Per-Hop Forwarding Behaviours in the DiffServ Internet. In: IEEE Symposium on Computers and Communications, Antibes-Juan les Pins (2001)
9. Rose, K., Regunathan, S.: Toward optimality in scalable predictive coding. IEEE J. on Image Processing 7, 965--976 (2001)
10. Neto, A., Cerqueira, E., Curado, M., Monteiro, E., Mendes, P.: Scalable Resource Provisioning for Multi-user Communications in Next Generation Networks. Submitted to IEEE Globecom, New Orleans, USA (2008)
11. Wong, T., Katz, R.: An analysis of multicast forwarding state scalability. In: 8th Int. Conf. on Network Protocols, Osaka (2000)
12. Shaikh, A., Tewari, R., Agrawal, M.: On the Effectiveness of DNS-based Server Selection. IBM Research Report (2000)