QoS and Session Signaling in a 4G Network

Rui Prior and Susana Sargento

Abstract— **This paper describes the QoS subsystem of a proposed architecture for a next generation network, supporting both legacy data transfer applications and multimedia services. This subsystem is based on the concept of brokers, which manage resources and perform admission control of flows. Three main different scenarios for the integration of application signaling and network resource reservation signaling are supported. The scenarios differ on the entity that triggers the QoS requests to the broker and the resource reservation, in order to better support all types of applications and the needs of different operators. Simulation work shows the performance of each scenario in terms of signaling delay and response to high loads and sudden peaks of calls. Some guidelines for system dimensioning are also provided.**

Index Terms—**4G networks, Quality of Service, signaling**

I. Introduction

Next generation wireless communication systems will handle diverse types of services, across different types of access technologies, allowing for the optimization of the coverage/performance/cost factor under very different utilization scenarios. Providing mobility across domains using different access technologies in a seamless way, with no perceived service degradation for the user, is a major requirement for the next generation networks.

The scalable support for end-to-end QoS in such a universal mobile and heterogeneous scenario is one of the main topics in networks research nowadays. This technical problem is further compounded by the complex telecom business, with multiple types of operators foreseen in the market with quite different dimensions, characteristics and business cases, providing from basic data transport to intelligent services and multimedia. Although the use of the IPv6 protocol as a convergence layer greatly simplifies the support for seamless mobility and QoS across heterogeneous networks, the provision of multimedia and value-added services in such multi-provider environments requires a common signaling framework for session negotiation, network resource reservation, and session and QoS renegotiation. This framework must integrate application signaling and resource reservation protocols in order to ensure that enough resources are available for a good user-perceived service quality, and that the use of those resources is authorized.

In this paper we present a performance study of a 4G network architecture, with emphasis on the QoS subsystem. This subsystem is based on the concept of QoS Brokers that manage network resources and perform admission control of flows. The architecture provides a large degree of flexibility regarding the interaction of applications and resource reservation/QoS signaling, giving rise to three different scenarios for session setup and (re)negotiation, differing on the entity that issues requests to the QoS.Broker: (i) the mobile terminal itself; (ii) a service proxy; and (iii) a module at the access router, able to perform application signaling parsing and modification. In [12] these signaling scenarios were presented and qualitatively compared with regard to session setup and negotiation, security and flexibility. In this paper, the efficiency of session setup in the different signaling scenarios is analyzed and compared through simulation, using ns-2 [1]. The results show that the scenarios have similar setup delays. We also analyzed the system's response to high signaling load conditions, from where we derive guidelines for system dimensioning in face of expected load and policies to be implemented in order to better handle load peaks.

The rest of the paper is organized as follows. Section II gives a brief overview of the architecture, focusing on the QoS-related elements and aspects. A description and brief analysis of the different signaling scenarios is presented in section III. Simulation results are presented and discussed in section IV. Finally, section V presents the main conclusions and suggests some topics for further work.

II. Architecture Overview

The main focus of 4G systems is the support of the aforementioned heterogeneity under a unified network architecture, allowing for an incremental development of new advanced services for the users. The IPv6 protocol is used as the convergence layer of the unified platform: IPv6 creates an abstraction layer that hides technology-specific parameters from advanced services. The native support of mobility in IPv6 is also of major importance for 4G communication systems. However, in order to provide completely seamless mobility, an extension based on fast handovers [2] is applied to IPv6. These issues and their relation with QoS aspects have already been addressed in the literature (e.g. [3]).

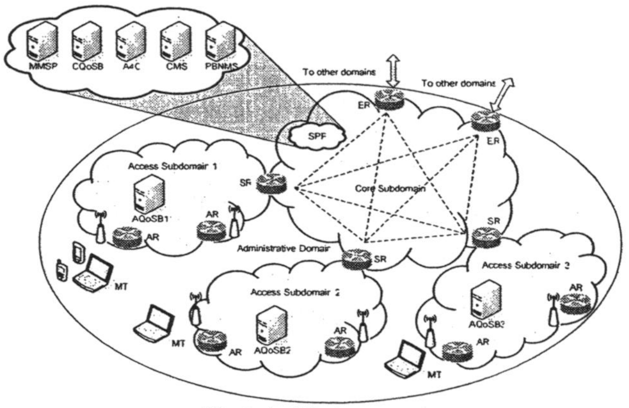

Fig. 1. Architecture overview

This work is based on results of the IST FP6 Integrated Project DAIDALOS, funded by the European Community's Sixth Framework Programme. It reflects the author's views, and the EC is not liable for any use that may be made of the information contained herein.

R. Prior (rprior@ncc.up.pt) is with the Laboratory of Artificial Intelligence and Computer Science – LIACC – University of Porto, Portugal.

S. Sargento (ssargento@det.ua.pt) is with the Institute of Telecommunications, University of Aveiro, Portugal.

1-4244-0000-7/05/$20.00 ©2005 IEEE.

Fig. 1 shows our proposed architecture for a next generation network. Each administrative domain may contain a number of access networks, each of them supporting several (wireless) access technologies, connected to each other and to different domains by a core network. This architecture allows for different operators to work in a common environment, with support for access services, other transport services, and advanced services. Operators can have special contracts between them – federation mechanisms – enabling a more integrated service to the end user.

QoS Brokers (QoSB) in the Access Network (AN) control the admission of new flows and the handovers, and manage network resources, configuring the Access Routers (AR) accordingly, in a PDP-PEP (Policy Decision/Enforcement Point) relationship. They also optimize the usage of operator resources by load balancing users and sessions among the available networks (possibly with different access technologies) through the use of network-initiated handovers. Contrary to the access, where IntServ-like [4] per-flow reservations are used for better control, QoS support in the core is based on the DiffServ model [5] for scalability. Though resource management is performed on an aggregate basis in the core and inter-domain segments of the path, information on the aggregates is propagated to the AN QoSB, where it is used for admission control in order to achieve end-to-end QoS. This combination of per-flow and per-aggregate processing in a two-layer hierarchy allows the architecture to provide fine-grained QoS control while keeping the scalability properties of per-aggregate core resource management, decoupled from per-session signaling.

The ARs contain advanced functions (Advanced Router Mechanisms - ARM [7]) that enable them to map application to network level QoS requirements, issue resource reservation requests to the QoSB and filter the QoS configurations in the application signaling messages of multimedia services using an out-of-band protocol (e.g., SIP [8]). A QoS client module in the Mobile Terminals (MT), capable of marking application packets for a QoS service and to issue requests to the broker, may also perform the resource requests.

In the core network (CN), there is a Service Provisioning Platform (SPP) able to provide services and applications on top of this network. A MultiMedia Service Platform (MMSP), consisting of a broker and proxy servers, is responsible for the provision and control of multimedia services. It is also capable of mapping application level QoS configurations to network resource requirements and of performing QoS requests for the flows. This architecture, thus, has a large degree of flexibility in QoS signaling, enabling the use of a diversity of QoS access signaling scenarios that fulfill the needs of the different applications and the business cases of different operators. Unification of the scenarios is achieved by the centralization of admission and handover control at the ANQoSB. The SPP contains a CNQoSB, responsible for resource management in the core. Policies for resource management are defined by the PBNMS (Policy-Based Network Management System) and sent to the CNQoSB, where they are cached in a local repository for use. The Central Monitoring System (CMS) collects statistics and other network usage data from network monitoring entities, and feeds the PBNMS and the QoSBs with this information for proper network resource management.

When a user registers in the network, the ANQoSB retrieves from the A4C (Authentication, Authorization, Accounting, Auditing and Charging) a subset of the user profile (to improve the network efficiency and scalability). This subset, termed NVUP (Network View of the User Profile), contains information on the set of network level services (classes of service, bandwidth parameters) that may be provided to the user, reflecting its contract with the operator. Similarly, a Service View of the User Profile (SVUP), containing information on the higher level services available to the user (e.g., voice calls, video telephony, and the respective codecs), is retrieved by the MMSP to control multimedia services.

Other proposals for 4G architectures have been made, e.g. [9] and [10], and, in fact, our work has been influenced by [3]. In broad terms, our architecture is more flexible with a fully integrated approach to IP-based communication with different types of applications and protocols, the customization/optimization of the architecture according to the expected service mix, and the integrated support of multiple QoS service models (defined by operator policies). New signalling methodologies are being thought for supporting QoS in mobile environments, like those being defined by the NSIS Working Group [11]. These approaches can be used both in distributed and centralized admission control, and may, therefore, be used in our architecture for signalling in the access network. This is a topic for further work.

III. Signaling Scenarios

In this architecture, MMSP, ARM and MT are the elements able to issue QoS requests. In this section we describe different scenarios with the initiation of a multimedia call between two terminals. Although the example uses SIP, other signalling protocols may be used, leading to message exchange sequences not differing much from these ones. Furthermore, the ARM and MT scenarios support services that do not use an out-of-band signalling protocol. A more thorough description of the scenarios is available in [12].

Fig. 2 illustrates the MT scenario in a simplified example of a multimedia session initiation, considering terminals in different domains. Mobile terminal MT1, through the respective QoS Client, maps the application requirements to network services and QoS requirements, and sends a request to its serving ANQoSB1 (via a QoS attendant at AR1) with this information. QoS signalling between the QoS Client and the attendant is implemented as an extension to RSVP [6], which is local between the MT and the AR; communication between AR and ANQoSB1 is based on the COPS

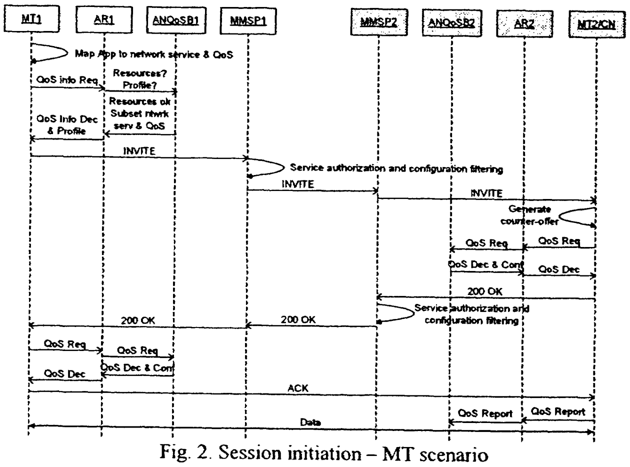

Fig. 2. Session initiation – MT scenario

protocol [13]. ANQoSB1 answers with information on the available services according to the user profile and the current network status. If allowed by ANQoSB1, MT1 sends an INVITE message with an initial offer of QoS configurations to MT2. When receiving the INVITE, MMSP1 performs service authorization, filtering out services not allowed by the SVUP. If the service is authorized, the INVITE is forwarded to the MT2. MT2 matches the QoS configurations in the INVITE to its own set, requests ANQoSB2 for available resources, and generates a counter-offer, included in the 200 OK message (the 180 Ringing message, not relevant for QoS, is omitted in the figure). On receiving this message, MMSP2 filters the services to those authorized. When the message arrives at MT1, it chooses the service to use, informs ANQoSB1 to configure the ARs accordingly with the required bandwidth and queue available space for the flows and classes, and sends an ACK containing the final configuration that will be used. This message triggers the sending of QoS reports to ANQoSB2, confirming the QoS configurations in the routers. In this scenario, applications without an out-of-band signalling protocol may also be made QoS-aware by coding them to invoke this procedure. Accounting processes are omitted in this paper, for simplicity.

In the MMSP scenario, the terminals do not perform QoS requests; they just perform SIP signalling through extended proxy servers, capable of parsing QoS configurations, mapping them to network resource requirements and contacting the brokers (using COPS) to perform the QoS requests. The proxies also enforce policies configured by the operators concerning the services allowed by the user contracts (reflected by the respective SVUPs).

In the ARM scenario, it is the AR that performs application to network level QoS mapping and issues resource reservation requests to the QoS Broker (again through COPS). Since the AR is always on the data path, legacy, QoS-unaware applications with in-band signaling only are equally supported by this scenario: for example, when the ARM sees a TCP SYN packet with destination port 23, it knows a Telnet service is being started and requests resources according to the operator's policy. The information needed to perform this action is supplied by the QoS Broker at boot-up of the AR. The information on general QoS profiles for legacy applications comes from the PBNMS, and reflects a mapping of operator business models into network policies. This scenario is preferred when a set of simple well-known services must be universally supported.

IV. Simulation Results

The efficiency of signaling for multimedia calls in the QoS signaling scenarios described in the previous section was evaluated using the ns-2 simulator [1]. We performed several experiments to evaluate the delay in establishing a session, as well as the response to congestion situations (establishment of a massive number of calls) in each of the signaling scenarios. The simulations comprise all possible combinations of (1) caller terminal at the home domain or roaming, (2) callee terminal at the home domain or roaming, (3) caller and callee physically attached to the same or different domains and, in the first case, (4) caller and callee physically attached to the same or different ANs, therefore representing all intra- and inter-domain call scenarios.

SIP is not implemented in the standard ns-2. Although a previous implementation from NIST [14] existed, it is incomplete and difficult to extend, and supports only stateless entities. Therefore, we have performed a new implementation of SIP, layered, with stateful entities, supporting user agents (UA) and proxies/registrars, and enhanced to support the QoS-aware UAs and MMSP; it also supports reliability of provisional responses (100rel) SIP extension [15], used in these simulations.

Processing delays in the elements are accounted for in the simulation models. Since the prototype implementation of these elements (for the testbed) is not yet complete, delay values were extrapolated from measurements in other elements performing similar tasks (e.g., MMSP delays were extrapolated from measurements on SIP proxies). Message processing is performed in a FIFO fashion, meaning that processing of each message can only begin after all previous messages have been processed. Each message type takes a fixed amount of time to process, which is different for the different message types. Processing delays for SIP messages were simulated at both the MT (10ms) and the MMSP (0.8ms), with an increment for messages with SDP bodies (10ms in the MT and 0.8ms in the MMSP); this increment is larger when the entity performs QoS Broker requests (15ms in the MT and 1ms in the MMSP). At the AR, processing delay is considered for SIP messages with SDP bodies (0.2ms), much larger in the ARM scenario (1ms). ANQoSBr request processing is also accounted for (1ms). The remaining processing delays are considered negligible when compared to these, and ignored in the simulations.

These simulations assume that the terminals are properly registered, meaning that valid NVUP and SVUP are already in place at the ANQoSB and the MMSP, respectively, allowing them to act as PDPs for network resources (ANQoSB) and multimedia services (MMSP). This assumption allows us to simulate post-paid call initiation scenarios where the accounting/charging messages are not in the critical path of the session setup signaling.

The message sequences are derived from those presented in section III, but use 100rel to avoid ghost rings (calls dropped as soon as the callee picks up the phone due to lack of resources). In the scenario where the MMSP issues the QoS requests, the caller starts by sending an INVITE with a configuration offer. The counter-offer is conveyed in a reliable "183 Session Progress" response, and the callee equipment starts ringing on receipt of its confirmation, after which there is a configurable random delay, corresponding to the time it takes for the user to answer the call, before the session is accepted. There are two possible sequences for rejected sessions: if the pre-reservation on the caller side fails, the MMSP of the caller immediately rejects the call with a "488 Not Acceptable Here" response; if it is the QoS request at the callee side to fail, the MMSP of the callee issues a CANCEL request to abort the session, resulting in a "487 Request Terminated" response from the callee UA.

Unlike the MMSP, the ARM is not a full-featured SIP entity. In particular, it does not generate new SIP messages. Therefore, when the ARM is responsible for QoS requests, if the initial request is rejected, the ARM does not generate a "488 Not Acceptable Here" SIP response. Similarly, if a full request is rejected by the ANQoSB, it does not generate a SIP CANCEL request; instead, it simply modifies the SDP body to indicate that none of the codecs is supported, relying on the SIP UAs on the MTs to react accordingly, abort-

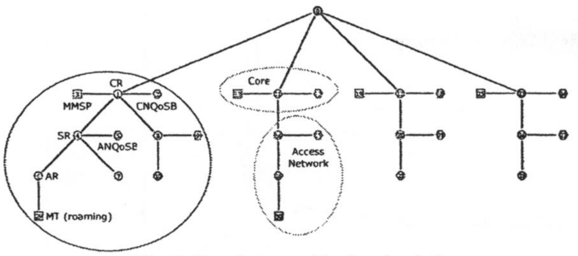

Fig. 3. Topology used in the simulations

ing the session. Therefore, although the sequence for a successful call is very similar to that of the MMSP scenario, the failure sequences have an additional round-trip time.

In the MT scenario, if the initial QoS request at the caller side fails, no SIP INVITE is ever sent; if the QoS request at the callee side fails, the session is immediately rejected with a "488 Not Acceptable Here" response.

Fig. 3 shows the topology used in these simulations, containing four domains, the leftmost one containing two ANs, one of which with two ARs. Although very simple, this topology allows us to simulate all possible combinations of roaming and non-roaming terminals: physically attached to the same AR, same AN and different AR, same domain and different ANs, or to different domains.

The implemented ANQoSB has topological knowledge of the bandwidth available in each of the interfaces of the ARs it controls. In the current version, however, it considers only access resources; core and inter-domain resource availability is not considered for admission control purposes at this stage (to be considered in the near future).

Some simplifications are assumed in the simulation model, namely the absence of DNS lookups and messages for the translation of home to care-of addresses at the MMSP. The latter is due to the existence of different alternatives to perform the translation, using the A4C or the Home Agent directly; these alternatives will be evaluated in further simulations. However, it must be highlighted that it would be possible to integrate the Home Agent and the MMSP, in practice dispensing with any external message exchange to perform the translation, similarly to these simulations.

The efficiency of call setup signaling is evaluated in the three QoS signaling scenarios for all possible combinations of intra- and inter-domain scenarios. To this end, 32 terminals are uniformly distributed among the different ANs, each terminal having a 50% probability of being at its home domain and 50% of being roaming; random calls are generated between pairs of terminals, with an average duration of 120s and a mean interval between call generation of 15s, for a simulated time of 24 hours. The roaming scenarios (relative locations of the terminals intervening in a call) are identified by four letters, *abcd*, where *a* indicates if the caller terminal is at its home domain (*a=h*) or roaming (*a=r*), *b* holds similar information for the callee, *c* indicates if the terminals are connected to the same administrative domain (*c=y* or *c=n*) and *d* if they are connected to the same AN (*y* or *n*). For example, *hryn* means that the caller is at home and the callee is roaming, both are attached to the same domain (that is, the callee is roaming at the caller's home domain) but to different ANs.

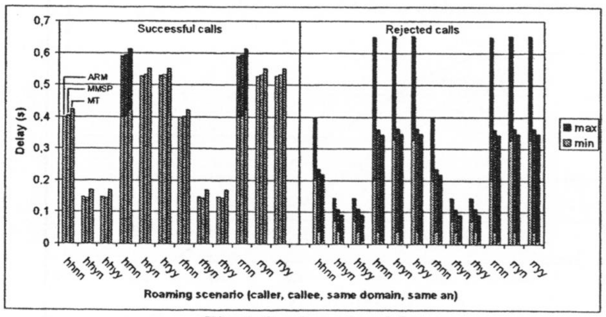

Fig. 4. Session setup delay

The session setup delay results in light signaling load conditions, simulated by using long calls and non-cumulative processing delays, are shown in Fig. 4. It is worth noting that while the availability of resources is related to the number of established calls, the signaling load is related to the number of calls being initiated or terminated, since signaling is performed only at call setup and teardown. These delays are those sensed by the caller, that is, from the instant the it begins the signaling to the instant it receives the final response (200 OK or 4xx) from the callee and responds with the ACK (the call answering delay is, obviously, subtracted from this value).

In successful calls, though there are slight differences between the signaling scenarios, they are of very little significance when compared to those imposed by the roaming scenarios: the dominant factor is the delay inflicted at each inter-domain link, of 30ms (roughly 6000km in optical fiber, ignoring router processing) in these simulations. Notice that the delay at the radio links, though potentially large, is common to all scenarios, thus not a discrimination factor. The most favorable scenarios are those where both terminals are physically at the home domain of the callee (*xhyx*), where no inter-domain links are traversed. When the callee is roaming, the initial INVITE and all its responses go through its home MMSP even if it is attached to the same domain as the caller, imposing a much larger setup delay; notice, however, that this does not apply to any further SIP requests, such as PRACK, ACK, BYE and possible re-INVITEs for session renegotiation. The re-INVITE case is particularly important, since we want to minimize disruption at handover time when the lack of resources at the new network imposes a renegotiation of the session.

The worst scenarios are those where the callee is roaming on a different domain than the caller, and the caller is not at the home domain of the callee (*xrnn max*). In this case two inter-domain paths are crossed by the INVITE and its responses; one inter-domain path is crossed by the other SIP messages. If the caller is at the home domain of the callee (*xrnn min*) the INVITE crosses only one inter-domain path.

Regarding rejected calls, the setup delays of the signaling scenarios are inverted, the ARM being much worse than the other two in most roaming scenarios. This stems from the fact that, since the ARM is not a fully featured SIP entity, it cannot generate new requests (CANCEL) or responses (e.g., 488 Not Acceptable Here), as described above. Here, the *min* values correspond to calls rejected at the caller side, and the *max* values correspond to those rejected at the callee side. Rejected calls are, however, a very small minority of the overall attempted calls, meaning that this factor has little relevance in the choice of a signaling scenario.

From the above presented results, particularly those for successful session setup (the most relevant ones), we conclude that the efficiency, in terms of delay, of the session setup procedure under light load is not a decisive factor in the choice of one of the different signaling scenarios for

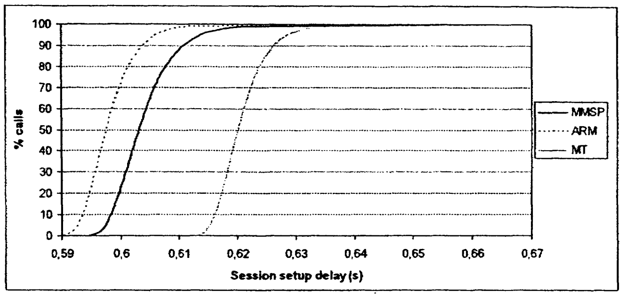

Fig. 5. Setup delay CDF (rrnn, 70 calls/sec)

SIP-based calls. In fact, except for the rejected sessions that take noticeably longer in the ARM scenario but are not much relevant since they will be infrequent, the three scenarios exhibit very similar signaling delays.

In all the scenarios, QoS requests are triggered by responses containing an SDP counter-offer as the message body. Such responses must be sent reliably, that is, they are either provisional responses requiring confirmation by means of a PRACK request, as in these simulations, or 200 OK final responses confirmed by an ACK. In either case, if no confirmation is received within a time interval (defaulting to 500 ms for the first time), they will be retransmitted. Care must be taken to avoid these retransmissions, since these are relatively large messages with a counter-offer in the body. If the summed delays from the transmission of the response to the reception of the confirmation, including all processing delays, cannot be consistently kept below 500 ms, this timer should be increased in real scenarios.

In a second experiment we evaluate the distribution of the setup delay of calls in the worst-case *rrnn* roaming scenario, under a medium/high offered signaling load of 70 new calls per second, with an exponential distribution of the time interval between generated calls. We used only 2 ANs in different domains, but a very large number of terminals (3000, 1500 in each AN/domain), in order to support the very large number of simultaneous calls. All the terminals are roaming and belong to the unused domains. The calls are initiated between a terminal attached to the first domain and another one attached to the second domain. Admission control at the ANQoSBs was set to always accept the requests.

Fig. 5 shows the results of this experiment by means of the Cumulative Distribution Functions of the setup delay in each scenario (e.g., in the MMSP scenario 20% of the calls are established in less than 0.6s). As can be seen, the setup delay does not vary much, even for the few percent calls where its value is larger. The largest measured setup delay exceeds the shortest one by 11% in the MMSP scenario, 9% in the ARM scenario and 8% in the MT scenario; for the 99% percentile, the values are 4%, 4% and 3%, respectively. These are average results of 5 simulation runs of 3600 seconds (corresponding to ca. 250000 calls each).

In another experiment we evaluated the limits of the system in the different signaling scenarios by increasing the offered load, in similar conditions to the previous one. The 99th percentile of successful call setup delays for the worst-case *rrnn* roaming scenario is plotted against the average number of generated calls in Fig. 6 (the reason for using the 99th percentile instead of the average will be explained later on). These results are also the average of 5 simulation runs. As can be seen, the setup delay, approximately constant up to a certain load, grows explosively after that value. This fact is explained by the transaction-stateful character of the MMSPs: at a given point, processing delays accumulate up to a sufficient value for the SIP retransmission timers to expire. Since retransmitted messages also take time to process, a snowball effect occurs, and delays become so large that calls start failing due to timeout of the INVITE transaction, not to ANQoSB rejection by lack of resources. Measures should be taken to avoid reaching this unstable state: load balancing between MMSP boxes must be dimensioned for worst-case expected load; additionally, a policy should be implemented in the MMSPs such that new requests are ignored (or summarily rejected) as soon as processing load exceeds a given threshold. It is worth noting that although the MMSP scenario is the first one to reach its limits, as expected since the MMSP is doing more work and is the bottleneck, values for the other scenarios are very close.

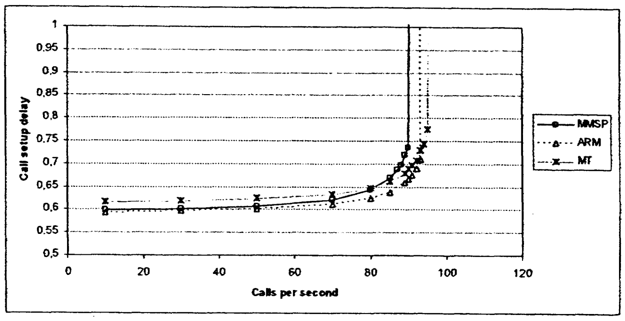

Fig. 6. Percentile 99 call setup delay vs. signaling load

In a last experiment we evaluated the system response to a sudden peak of calls. We used the same *rrnn* roaming scenario but initiated calls at deterministic generation rates: first, a rate of 10 new calls per second for 100 seconds; then, a peak rate of 200 new calls per second for 5 seconds; lastly, we restored the initial rate of 10 new calls per second. The peak of calls causes processing congestion in the MMSP, triggering the aforementioned snowball effect. However, since the call generation rate after the peak is quite low, the system is able to come back to a stable operation state. In order to evaluate how long it takes for the system to recover, we plotted the call setup delay against the session initiation instant of the calls in the three signaling scenarios. The results are shown in Fig. 7. As may be seen, the peak causes very large, unacceptable delays in all scenarios; however, these delays reach higher values and take longer to recover in the MMSP scenario than in the other two. It is worth noting that the results for the ARM and MT scenarios are almost overlapping, since they have the same amount of processing in the bottleneck component, the MMSP.

As the frequency of new calls in the steady state (before and after the peak) increases, the time it takes for the system

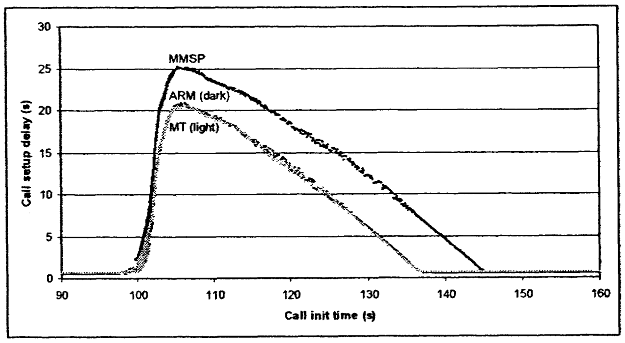

Fig. 7. System response under a peak of new calls

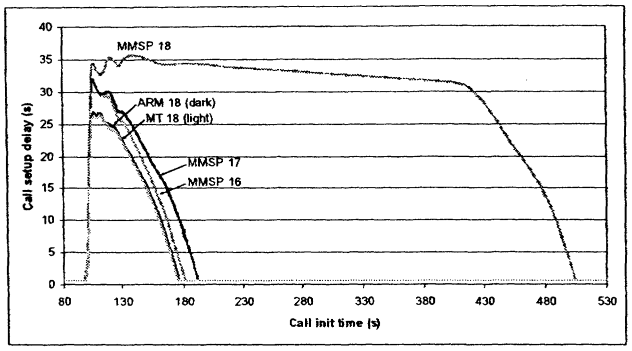

Fig. 8. Peak of new calls – varying steady-state load

to recover after the peak of calls increases. Eventually, the time to recover from the peak rises dramatically; this effect is illustrated in Fig. 8, where three curves are shown for the MMSP scenario, corresponding to steady-state call initiation rates of 16, 17 and 18 calls per second. In the last case, we may see that there is a slow decay segment in the setup delay curve until a certain point is reached, where the curve begins to decay with a similar pattern to the other cases. For the sake of comparison, curves for 18 new calls per second are also shown for the ARM and MT scenarios, from where we may observe that the critical steady-state load for these scenarios has not yet been reached.

In Fig. 9 we plot the recovery time from the peak of calls against the steady-state call generation rate for the three signaling scenarios. It may be seen that the recovery time is approximately the same in the ARM and MT, and higher in the MMSP scenario. Additionally, in the last one there is an explosive growth from 17 to 18 new calls per second in steady-state; this growth is more gradual in the other two scenarios, suggesting that the MMSP scenario is somewhat less stable regarding overloads than the others.

In face of the results described in the previous paragraphs, the reason for using the 99th percentile instead of the average in Fig. 6 should become clear. With a sufficiently high number of calls per second, the system cannot recover from the snowball effect, and further calls are rejected. By using the 99th percentile we capture the effects just before calls start being rejected due to timeouts; with average values, this effect would have been masked out by the large number of calls previously established with very low delay. The average of successful and failed calls (not shown due to space limitations), however, exhibits a similar effect (even more dramatic) than the 99th percentile of accepted calls.

The behavior of the system with respect to call setup delay during and after a peak of very high load suggests that the steady-state offered signaling load must be way below what can be handled, on average, by the MMSP, in order to have enough processing slack to absorb the snowball effect of SIP retransmissions. Once again the interest of a policy of summary rejection of calls at the MMSP when the signaling load exceeds a given threshold is demonstrated: the system would recover much faster from peaks, allowing for a higher average steady-state load, which translates in less cost in hardware for similar load resilience characteristics.

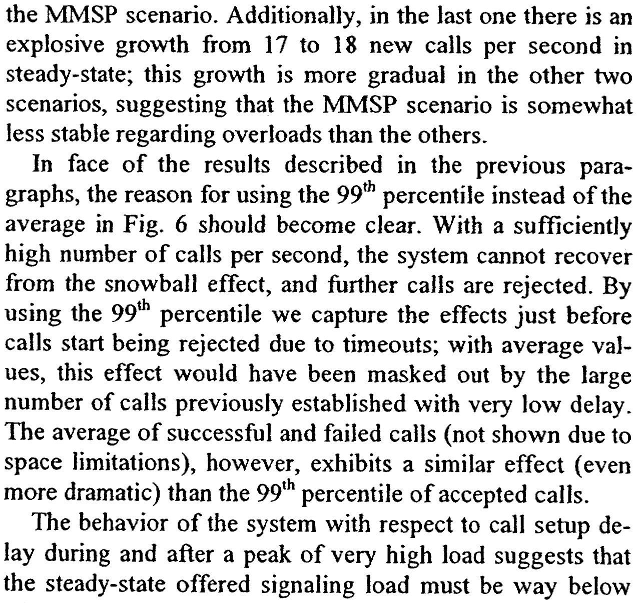

Fig. 9. Recovery time from the peak of calls (200/s for 5s)

V. Conclusions

This paper presented a simulation study of a 4G communication system based on IPv6. We presented and analyzed different scenarios for interaction between the QoS Brokers and the other QoS-related entities – centered on the terminal (PDA, intelligent cellular phone), on service proxies (e.g. SIP proxies or application servers), or in the access routers – and the corresponding strategies for the interaction between application- and network-level QoS signaling. Through a number of simulation experiments, we compared the behavior of the different scenarios. The results indicate that under normal operating conditions the efficiency of the different scenarios is comparable; under heavy load conditions the MMSP scenario exhibits problems before the other ones, an expected result since more functions are performed by that element. The fact that excessive signaling load problems are greatly exacerbated by the snowball effect of SIP retransmissions means that a policy of summary rejection of new calls when the processing load reaches a certain threshold at the proxy should be implemented to prevent this effect and, thus, improve the resilience to signaling overload.

As further work, we intend to evaluate the behavior of the system regarding adaptive applications, for which renegotiation may be performed at handover time. As previously mentioned, we also intend to evaluate the use of NSIS for QoS signaling in the access network.

References

[1] http://www.isi.edu/nsnam/ns/

[2] R. Koodli (ed.), "Fast Handovers for Mobile IPv6." Internet Draft, Oct. 2004.

[3] V. Marques, R. Aguiar et al., "An IP-Based QoS Architecture for 4G Operator Scenarios." IEEE Wireless Communications Magazine, vol. 41, issue 3, Mar. 2003, pp. 120-124.

[4] R. Braden, D. Clark and S. Shenker, "Integrated Services in the Internet: an Overview." IETF RFC 1633, Jun. 1994.

[5] S. Blake et al., "An Architecture for Differentiated Services." IETF RFC 2475, Dec. 1998.

[6] R. Braden (ed.) et al., "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification." IETF RFC 2205, Sep. 1997.

[7] D. Gomes, P. Gonçalves and R. Aguiar, "QoS Transignaling for Heterogeneous Networks." In Procedings of the 11th International Conference on Telecommunications – ICT'2004, Ago. 2004.

[8] J. Rosenberg et al., "SIP: Session Initiation Protocol." IETF RFC 3261, Jun. 2002.

[9] D. Wisely, E. Mitjana, "Paving the Road to Systems Beyond 3G - The IST MIND Project." Journal of Communication and Networks, Dec. 2002.

[10] Joachim Hillebrand, et al., "Quality-of-Service Signaling for Next-Generation IP-Based Mobile Networks." IEEE Communications Magazine, June 2004, pp. 72-79.

[11] Next Steps in Signaling (NSIS) working group charter, http://www.ietf.org/html.charters/nsis-charter.html

[12] R. Prior, S. Sargento, D. Gomes and R. Aguiar, "Heterogeneous Signaling Framework for End-to-End QoS Support in Next Generation Networks." In Proceedings of the 38th Hawaii Conference on System Sciences (HICSS). Jan. 2005.

[13] D. Durham (ed.) et al., "The COPS (Common Open Policy Service) Protocol." IETF RFC 2748, Jan. 2000.

[14] http://snad.ncsl.nist.gov/proj/iptel/

[15] J. Rosenberg and H. Schulzrinne, "Reliability of Provisional Responses in the Session Initiation Protocol (SIP)." IETF RFC 3262, Jun. 2002.