# INQA: InterNetwork QoS Agreements
## A New Protocol for Dynamic SLS Control in Next Generation Networks

Ioannis Psaras and Lefteris Mamatas
Demokritos University of Thrace
Vas. Sofias 12, 67100 Xanthi, Greece
Email: {ipsaras, emamatas}@ee.duth.gr

Paulo Mendes
DoCoMo Euro-Labs
Landsbergerstr 312, 80687 Munich, Germany
Email: mendes@docomolab-euro.com

***Abstract*— Although Internet packet routing is suitable to provide best-effort data transport, the control of Quality of Service (QoS) is needed for data traffic with extra quality requirements. Since there are already possible solutions for the provision of IP networks, our work focuses on the lack of a dynamic approach to control QoS between networks. With this goal in mind, we design a new signaling protocol to control inter-network QoS agreements, each of which is defined as one Service Level Specification (SLS). We implement the proposed protocol in an experimental test-bed and evaluate it in a scenario with moving networks. Our findings show a good performance of the proposed protocol in terms of its relative scalability and convergence time during the movement of networks.**

## I. INTRODUCTION

The Internet control plane enables packet routing between networks, which makes it suitable to provide best-effort data transport between an increasing number of hosts. Regarding data traffic with extra quality requirements, more advanced features are needed to control *Quality of Service* (QoS) between hosts. Currently there are a number of possible solutions to perform the provisioning of networks, from static-provisioning to over-provisioning, passing by signalled-provisioning. Independently of the mechanism to control QoS inside networks, a major limitation for the end-to-end control of quality in heterogeneous[1] IP environments is the lack of a dynamic approach for inter-network QoS control.

This situation tends to get worse in a future scenario in which networks are mobile, as happens with public transports where passengers would like to participate in broadcast multimedia communication sessions. This scenario, analyzed in the IST Ambient Networks project [7], brings extra requirements to inter-network QoS control, which must be automatic due to the dynamic behavior of networks, and at the same time shelter the complexity brought by network heterogeneity.

Two signaling paradigms are seen as possible starting points to find a suitable solution to control inter-network QoS for different types of traffic: signaling based on flows and signaling based on *Service Level Specification* (SLS). The latter has an embedded aggregation method, by defining SLSs for different types of traffic.

Currently the *Internet Engineering Task Force* (IETF) is standardizing a flow-based signaling protocol (QoS-NSLP) [6]. In this paper, we present a new SLS-based signaling protocol to control *Inter-Network QoS Agreements* (INQA), whose pro-active behavior may bring more benefits to dynamic scenarios. To assess the extend of such benefits, we compare INQA with QoS-NSLP in an experimental testbed representing a public transport scenario setup. We find that INQA significantly reduces signaling overhead and communication interruption time upon a handover, but presents some scalability problems in terms of memory state requirements.

The remainder of this paper is organized as follows. In section II, we briefly analyse the advantages and disadvantages of flow-based and SLS-based signaling approaches. In section III we present in detail the proposed SLS-based signaling protocol. In section IV, we describe the prototype implementation of the two signaling protocols, and in section V we present our experimental findings. Finally, in section VI we conclude our analysis and enumerate some open issues for future work.

## II. BACKGROUND ON SIGNALING APPROACHES

The control of inter-network QoS in heterogeneous and dynamic environments requires inter-network signaling to allow the mapping of the QoS assurances provided by each network. In this section we cluster possible signaling solutions into two groups. One providing flow-based signaling and another group encompassing solutions that are traffic oriented, or better say solutions that provide SLS-based signaling.

In general, we characterize flow-based signaling approaches as the ones with control messages related to N-tuples, which define the network paths taken by data flows. Thereupon, devices are needed in the data path to configure network resources for a flow (aggregated or not). This type of signaling is implemented in protocols such as RSVP [2], QoS-NSLP, or the *Simple Interdomain Bandwidth Broker Signaling* (SIBBS) [4].

SLS-based signaling approaches are characterized as the ones allowing two adjacent networks to establish and maintain a set of bi-lateral QoS agreements (SLSs) for different types of traffic. This type of signaling is implemented in the proposed protocol (i.e. INQA) as well as in the *QoS extensions to BGP* (qBGP) [1], developed in the IST Mescal project.

Flow-based and SLS-based approaches may support different intra-network QoS control technologies by using a path-decoupled approach. This approach brings the benefit of allowing the use of different intra-network QoS control

---

[1] The term heterogeneous refers to networks with different capabilities and different resource control mechanisms.

technologies, by providing a clear separation between inter-network signaling and the signaling used inside a network to configure edge-to-edge data paths. On the other hand, flow-based path-coupled signaling approaches are not so suitable since they require all network devices, within all networks, to support the same signaling scheme.

Flow-based and SLS-based approaches may keep an updated state of the QoS established between networks. Nevertheless, although some flow-based approaches are being adapted to react to the movement of hosts, as is the case for QoS-NSLP, in general they have problems in handling the movement of networks, since they are only aware of mobility near flow initiators and destinations. Moreover, the movement of hosts requires flow-based approaches to signal the complete new path, since resources are coupled with N-tuples that are different before and after movement. SLS-based approaches are aware of mobility, as routing changes, at any network edge, and require only the adjustment of a sub-set of SLSs due to network movement.

Moreover, SLS-based approaches may furnish networks with knowledge about the SLSs that their neighbors may provide to them, since these approaches include control messages allowing networks to advertise service availability to neighbors. This characteristic will help to decrease the handover latency, since moving networks can be aware of the QoS level of communication services (SLSs) available in neighbor access networks. On the other hand, flow-based approaches are based on query or reserve messages that are triggered by a local need for resources, without having any previous knowledge about what QoS neighbor networks can offer. This lack of information about how to reach different networks, via different neighbors, while keeping certain QoS assurances even before the creation of data flows, brings further disadvantages to multi-homed hosts and networks.

Based on these differences, we endeavor to analyse the behavior of SLS-based protocols, which seem to bring advantages for next-generation networks. Since SLS-based protocols should be used in different inter-network scenarios, we state that INQA brings more advantages than qBGP, because the latter is tightly coupled with an inter-network routing protocol (BGP) [8]. Therefore, in the remainder of this paper we describe and evaluate INQA functionality.

## III. INQA: THE PROPOSED SIGNALING PROTOCOL

According to INQA, a network can have one out of the three following roles: provider, customer, and customer-provider. In the provider role, networks advertise local SLSs (which may be defined for different sub-networks), whereas in the customer role they negotiate SLSs, which may be used by local applications. Networks operating in the customer-provider mode want to resell SLSs advertised by their neighbors.

INQA may be implemented as a signaling layer protocol, on top of a generic signaling transport protocol, as happens with QoS-NSLP. The INQA protocol controls state in adjacent networks, but in contrast to QoS-NSLP, INQA does not use peer-to-peer refresh messages as the primary state management mechanism. Instead, the state is controlled based on the validity of the negotiated SLSs, which means that in the absence of an explicit release request, the SLS state is kept until it expires. Nevertheless, this is not a disadvantage in the case of changes in inter-network routing, since INQA peers are aware of the connections to the adjacent networks with whom SLSs were negotiated. To control the SLS state, INQA uses four message types: Advertise, Negotiate, Acknowledge, and Monitor. The Advertise message is used to announce to a set of neighbor networks information about SLSs specifying a certain QoS level to stipulated traffic with a specific scope. The Negotiate message is used to create and remove SLS state, based on which customer-networks are allowed to send a certain amount of traffic with a defined scope. The Acknowledge message is used to inform about the result of a previous Negotiation message. The Monitor message is used to allow a customer network query a provider network about the level of QoS assigned by previously negotiated SLSs[2], and to allow a provider-network notify a customer-network about the need to adjust an SLS, due to its under-utilization, for instance.

INQA supports receiver-driven, sender-initiated QoS agreements. Receiver-driven, since the signaling process is started when provider-networks advertise a set of SLSs. Sender-initiated, since the establishment of SLSs and the consequent reservation of resources is initiated by customer-networks. Advertisement messages travel in the opposite direction of the data traffic that is being signalled for. More precisely, to establish an SLS the provider-network includes an SLS object within an Advertisement message sent to one or more INQA adjacent peers. Based on their role and on the local application requests, the INQA peer generates a Negotiation message which is sent back to the provider-network, in order to establish a sub-set of SLSs. The provider-network will admit such requests if the advertised SLSs suffice to fulfill them, in terms of their geographical scope and amount of offered resources. Besides the geographical scope, the provider-network controls also the temporal scope of the negotiated SLSs, since besides their validity they may also be defined for specific time-periods, allowing the provider-network to multiplex SLSs over time. The negotiation process is terminated by the provider-network, which sends an Acknowledgment message to the Negotiator network. If the Negotiator network has customer-provider permissions, it can use the negotiated SLS to further advertise communication services to other neighbor networks. This re-advertisement of services allows the creation of chains of SLSs, and hence the implicit establishment of end-to-end QoS assurances.

Figures 1 and 2 illustrate the operational differences of QoS-NSLP (the standardized path-coupled version and the path-decoupled IETF study item) and INQA in a scenario with a moving sender. One of the major differences is the time at which operations occur. The operation of QoS-NSLP is initiated by a sender request, while with INQA operations

---

[2] The use of the Monitor message in untrusted environments needs further investigation, namely to understand how can a network check the answer provided by a neighbor network.

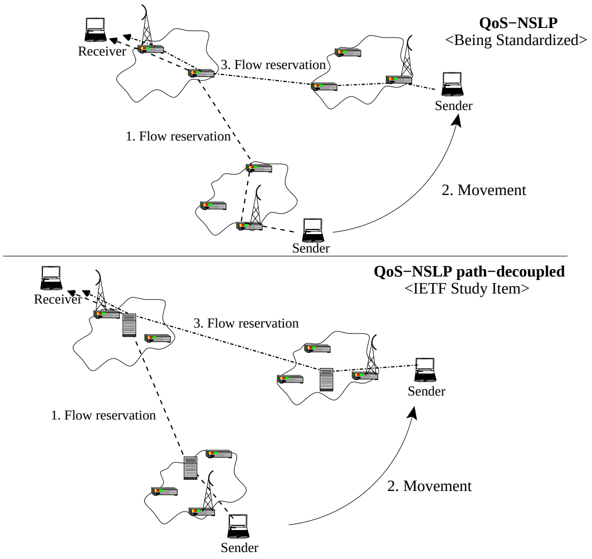


Fig. 1. Illustration of QoS-NSLP Signaling

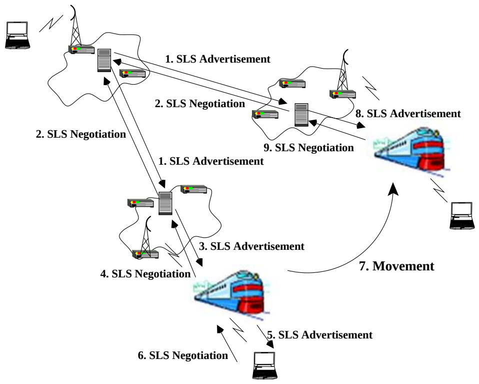


Fig. 2. Illustration of INQA Signaling

start by an advertisement sent by the network providing access to receivers. This means that INQA operates in an earlier-time space than QoS-NSLP. With INQA, a set of SLSs is established between networks, with the appropriate resource allocation inside each network[3], before the sender's request. When such request occurs, INQA signals between the sender and its local access network, while QoS-NSLP signals end-to-end. The situation is similar when the sender moves, in which case, INQA signals between the moving sender and its new access-network. The signaling (Advertise - Negotiate - Acknowledge) needed to establish a suitable SLS is triggered when the access-network detects the attachment of a new device. Since SLSs have a defined validity, the sender and the access-network may decide to keep them, allowing their re-use

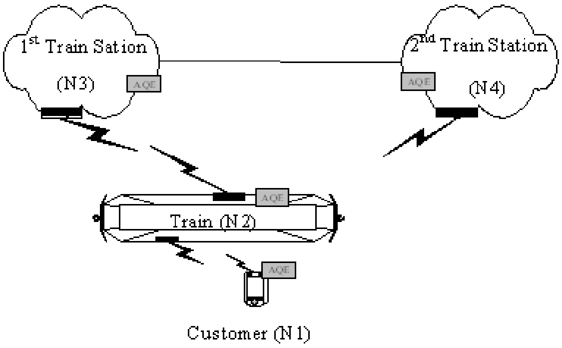


Fig. 3. Test-bed scenario

in case of a pre-scheduled movement, such as with a train. In the case of QoS-NSLP, the sender has to signal (Reservation - Response) across the complete new path until the destination to request the needed network resources for each on-going flow (aggregated or not).

## IV. EXPERIMENTAL SETUP

From the conceptual point of view, INQA seems to bring advantages in terms of handling network mobility and allowing networks to be aware of available SLSs in their neighbors. Nevertheless, it is not clear how INQA behaves with an increasing number of networks and SLSs, as well as with respect to the communication interruption time in a scenario with moving networks. Therefore, we analyse INQA in an experimental test-bed that includes a sender (N1), placed within a train network (N2), which moves between two access networks (N3 and N4), one of which encompasses a receiver. This setup is illustrated in Figure 3. From the flow-based approaches, we selected QoS-NSLP as a comparison benchmark, since it is in the IETF standardization track, where it is being extended to work also based on path-decoupled signaling.

The motivation behind the test-bed is to gather real-time measurements for INQA and QoS-NSLP. Since our study focuses on inter-network operations, each network is represented by a single network-node in the test-bed. All network-nodes are Linux IPv6 boxes. The movement is emulated by the *NRL Mobile Network Emulator* (MNE) [3], which selectively blocks different MAC addresses to generate different topologies. For instance, connections are blocked between two networks that are supposed to be out of each other's range in a specific topology. The control of routing is done by the *Optimized Link State Routing* protocol (OLSR) [5], which provides a flat-addressing space in the absence of a general accepted mobility management mechanism for moving networks.

On the test-bed described above, we implemented a prototype of QoS-NSLP and INQA. The former was implemented to operate in a sender-initiated and path-decoupled manner. Path-decoupled signaling was implemented assuming that neighbors are well-known, due to the fact that QoS-NSLP does not yet specify a peer discovery mechanism. Moreover, since QoS-NSLP does not support moving networks, we developed a

---

[3] Network resources are allocated to classes of traffic and not to specific flows. In this case resource wastage may be avoided by the monitor mechanism: a provider-network can collect information from local measurement mechanisms and notify a customer that it should update its negotiated SLSs. In the absence of a response, the provider may decide to re-allocate some of the resources previously allocated to that customer.

connectivity module that probes for connectivity changes in the MNE and signals both QoS-NSLP and the application about possible handovers.

In what concerns INQA, the advertisement and negotiation mechanisms are considered in this evaluation. Due to space limitations, we excluded from the current work the evaluation of the SLS monitoring mechanism, used to adjust the allocation of network resources.

QoS-NSLP and INQA are developed in Python, an object-oriented platform-independent language, which allows rapid prototyping and provides rich networking libraries. However one problem is that Python does not have an equivalent to the *sizeof* method found in the C programming language. Therefore, we measure the memory-state of QoS-NSLP and INQA using an indirect method: we convert each state-database to a binary representation and measure the amount of used bytes. This method produces more bytes than what is actually used, but it can be used for a relative comparison between the two proposals. The grammatical structure of the signaling messages is represented by an XML schema template, which is easily manageable and reduces the development time (due to the wide use of the XML parsers-analyzers). Moreover, to increase the fairness of the evaluation process we assume that the QSPEC used by QoS-NSLP, and the SLS used by INQA, encompass the same traffic profile and QoS related information for inter-network control.

The experimental setup is completed with two evaluation scenarios. In the first scenario all networks are stationary and we compare the behavior of INQA and QoS-NSLP in what concerns their signaling overhead and memory-state to control QoS between networks. This comparison is done while increasing the number of connected networks, the duration of the application, and the number of available network services. That is, initially we apply a configuration with a unique available service in N4 (provider-network) and an increasing number of networks, by attaching first N3 to N4, then N2 to N3 and finally N1 to N2. After reaching a stable condition, we evaluate the QoS-NSLP and INQA effort to keep the state alive, followed by a scenario where we increase the number of available services by one for each network.

In the second scenario, N2 (train), and consequently N1 (customer), move repeatedly between N3 and N4. In this scenario we compare the behavior of INQA and QoS-NSLP in what concerns their signaling overhead and communication interruption time until the data transmission is resumed after the handover. To evaluate the performance of the two signaling approaches the test-bed was configured with four networks, one service available in N3 and one flow from N1 to N3.

## V. EXPERIMENTAL EVALUATION

The evaluation of the experimental results is done based on the two scenarios described in section IV. In section V-A, we investigate the behavior of QoS-NSLP and INQA in a scenario with stationary networks, and section V-B encloses our analysis related to the scenario with moving networks.

### A. Scenario with Stationary Networks

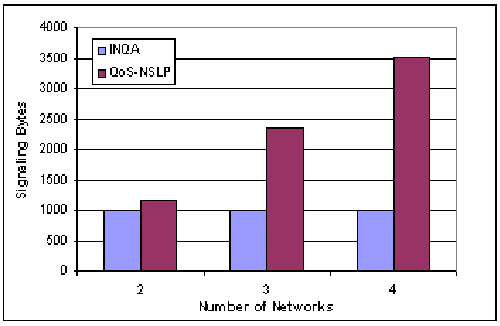
Each experiment focuses on a unique variable, namely the number of networks, duration of the application, and the available services.

*1) Increasing Number of Networks:* In what INQA is concerned, this experiment starts by N4 advertising its local service to N3 as soon as they get connected and N3 responding with the corresponding negotiation message. Since N3 has the role of customer-provider, it 're-sells' the available service to N2 as soon as they get connected, but only after adjusting the additive parameters, e.g. the offered delay. This process is repeated when N1 connects to N2. Finally, N1 (customer network) executes an automatic negotiation of the offered SLS, since it is configured to do so for SLSs that are suitable for well-known traffic, such as *Video-on-Demand* (VoD). Alternatively, N1 could negotiate several smaller SLSs according to the specific demand of its local applications. In this case, the bandwidth of each negotiated SLS would always be smaller than the bandwidth announced by N2 for the SLS to reach N4. After the negotiation phase between N1 and N2, any application in N1 may start streaming packets to any destination in N4, being the packets marked with the identifier of the negotiated SLS.

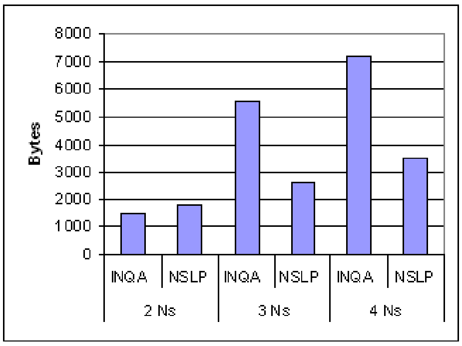
In what concerns QoS-NSLP, we consider that signaling may refer to a single flow of a flow-aggregate. In the latter case, we consider that the flow-aggregate belongs to the same pair of hosts (same IP addresses and different ports). The reason is as follows: QoS-NSLP describes signaling support for flow aggregation, but does not define neither how reservations are aggregated nor how end-to-end properties need to be computed. Moreover, any proposals to aggregate QoS-NSLP flows into the same flow-aggregate for nodes placed at network edges, including an indication of how de-aggregator QoS-NSLP nodes may collect information about aggregated flows, may deteriorate QoS-NSLP overhead.

Hence, with QoS-NSLP, every time two networks connect, no signaling is automatically generated to configure QoS. Instead, every time N1 wants to send a flow (aggregated or not) to a host in N4, QoS-NSLP needs to reserve resources for that flow in all intermediate network-nodes until the target host in N4. This means that QoS-NSLP exchanges two messages (Reserve and Response) between any two network-nodes, to reserve resources for each demanding flow.

Figure 4 a) shows that although INQA sends three inter-network messages to advertise, negotiate and acknowledge QoS agreements, instead of the two QoS-NSLP messages that pass between any two networks, it induces less signaling overhead than QoS-NSLP in terms of signalled bytes. This is due to the fact that the QoS-NSLP reserve message includes extra parameters required for its soft-state and multiple-hop functionality. We observe, in Figure 4 a), that INQA induces 1000 bytes of signaling overhead for any new network, while QoS-NSLP's overhead between any two networks increases directly proportional to the number of networks in the data path, since it performs end-to-end signaling. This experiment



(a) Signaling Overhead



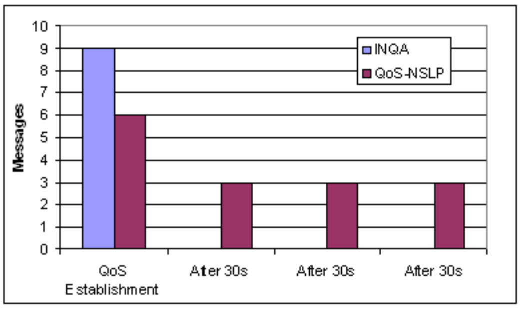
(b) Memory-state

Fig. 4. Signaling overhead and memory-state with an increasing number of networks

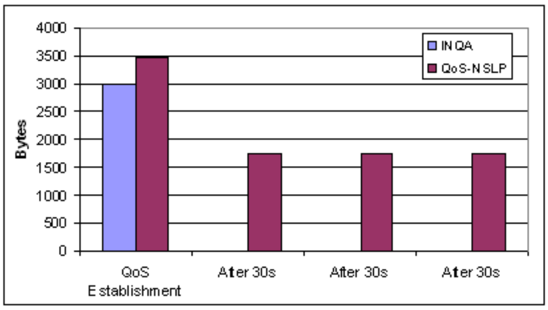
shows that INQA induces approximately 40% less signaling overhead than QoS-NSLP with an increasing number of networks.

The same Figure (Figure 4b)) captures also the memory-state required by each of the two approaches, in a situation with two, three and four networks. As might be expected, the policy of INQA to *establish and maintain/guarantee* the state of QoS agreements requires a larger memory-state than QoS-NSLP, which only *instantaneously reserves and periodically refreshes* the state of the path.

More precisely, the major difference between INQA and QoS-NSLP is that in the case of INQA the memory-state required to store SLSs depends on the role of the network. For instance, in this experiment N3 operates as a customer-network in a topology where only N3 and N4 are present, while it serves as a customer-provider network when N2 is also included. In the former case, N3 needs to store only the SLS negotiated as a customer of N4. In the later case N3 negotiates one SLS as customer of N4, and advertises that SLS, after applying the appropriate adjustments (e.g delay), to N2. Hence, N3 keeps information about two SLSs although those SLSs are practically the same (a more efficient method to store SLSs is being investigated). In case of a simple topology with two networks (N3 and N4), Figure 4 b) shows that QoS-NSLP requires more memory-state to store a QSPEC for one flow, than INQA needs to store one SLS (independently of the SLS scope, which can be set for one host in N4 or for the complete N4 network). This is due to the fact that, as mentioned before, a QSPEC object is slightly bigger than an SLS, since it includes the same QoS guarantee parameters as an SLS, but also needs to assign extra parameters to control the multi-hop operation of QoS-NSLP. Moreover, it is expectable that the memory-state required by QoS-NSLP will increase with the number of flows. In contrast, this is not expectable in case of INQA, if the demanding flows fit within the established SLS. Nevertheless, further investigation will be performed when aggregation mechanisms (for flows in QoS-NSLP and for SLS in INQA) are considered.



(a) Number of Messages



(b) Number of Bytes

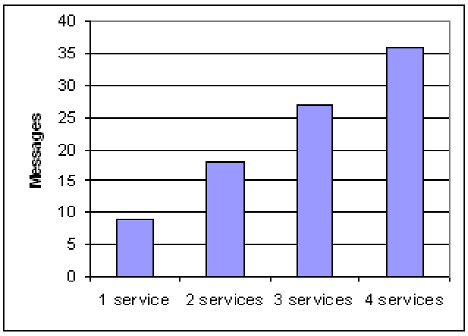
Fig. 5. Signaling overhead (number of messages and bytes) due to state refreshment mechanisms

*2) Stable System:* In this section we investigate the impact of state refreshing policies, for a scenario with one provider-network (N4), two customer-provider networks (N2 and N3), one customer-network (N1) and one flow between N1 and N4. As described above, INQA needs nine messages to set QoS agreements between all networks, while QoS-NSLP needs six messages to reserve resources in the end-to-end path for the considered flow.

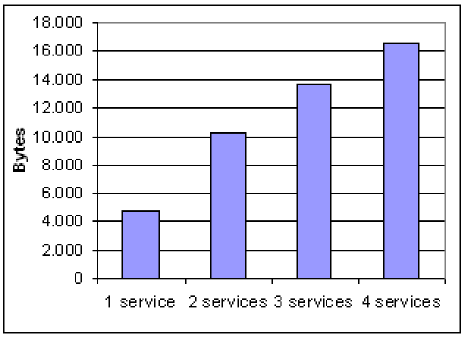
Regarding the maintenance of SLSs, as already mentioned, INQA does not require extra signaling messages to refresh them. Therefore, in our experiment INQA does not induce any extra signaling overhead after the SLS establishment.

On the contrary, QoS-NSLP refreshes the state reserved for each flow every 30 seconds, by sending a 'refresh' Reserve message from N1 towards N4, through all intermediate networks. This process adds extra signaling overhead, which is directly proportional to the duration of the flow.

Figure 5 shows the overhead (number of signaling messages and bytes) needed by QoS-NSLP and INQA to set and refresh



(a) Signaling Overhead



(b) Memory-state

Fig. 6. Signaling overhead and memory-state with an increasing number of available services

the network state. In this experiment, INQA transmits 40% less signaling messages than QoS-NSLP. The difference between the number of messages and the amount of bytes owes to the different sizes of INQA and QoS-NSLP messages, as explained before.

*3) Increasing Number of Available Services:* While in previous experiments only one network offered a service (SLS), in this experiment we analyse the behavior of INQA with an increasing number of available SLSs. For this purpose, we start the experiment with N4 offering one SLS, and progressively activating one extra SLS in each of the remainder networks (N3, N2, and N1). In addition, the customer-provider N2 and N3 networks always re-sell all SLSs advertised to them. This experiment leads to the exchange of 36 INQA messages and to the storage of around 16.000 bytes, when all networks have an available service to offer (see Figure 6). Although the situation corresponds to a worse-case scenario (see section V.B.2), since customer-provider networks always re-sell all SLSs advertised to them, it is clear that INQA has scalability problems in the presence of an increasing number of available services. Hence, further investigation is needed, in scenarios with different topologies and SLS distribution, to understand the kind of SLS aggregation mechanism needed in order to improve INQA scalability, in case of increasing number of available services. Nevertheless, it is worth to say that handling SLSs (types of traffic) brings higher scalability than handling flows.

### B. Scenario with Moving Networks

To analyse QoS-NSLP and INQA performance in a mobile environment, we use the scenario shown in Figure 3, where an end-user (N1) is inside a train (N2), and both move repeatedly between two (static) train stations (N3 and N4), performing a ping-pong movement. We use an FTP application, with flows from N1 to N3, assuming that flows have the same duration. Ten handovers are performed in this experiment, five from N3 to N4 and five from N4 to N3. After each handover the FTP application resumes its transmission from N1 to N3, which operates as customer and provider-networks, respectively.

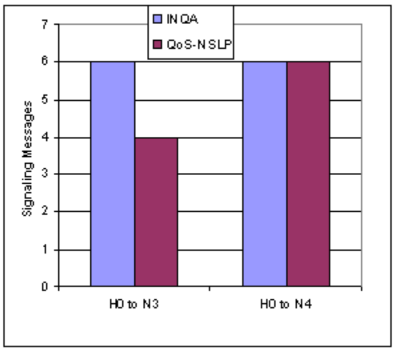
In the beginning of the experiment N4, acting as a customer, receives an SLS advertisement from N3, based on which it negotiates with N3 a corresponding SLS. Once N2 comes into the range of N3, the latter advertises to it the available SLS. Operating as a customer-provider, N2 establishes an SLS as a customer of N3 and another as a provider of N1. Using the established SLSs, one FTP application on N1 transmits data to a host in N3.

During the experiment N2 handovers from N3 to N4, from which it receives an advertisement message with an SLS to reach N3. After the establishment of an SLS between N4 and N2, the latter offers a similar service to N1 only if this service provides less QoS assurances than the SLS that N2 advertised to N1 before the handover. Otherwise, N2 and N1 use the SLS established before the handover. In both cases, the chain of SLSs between N1 and N3 allows the FTP application in N1 to resume the transmission to the same host in N3 with the same QoS assurances as before the handover.

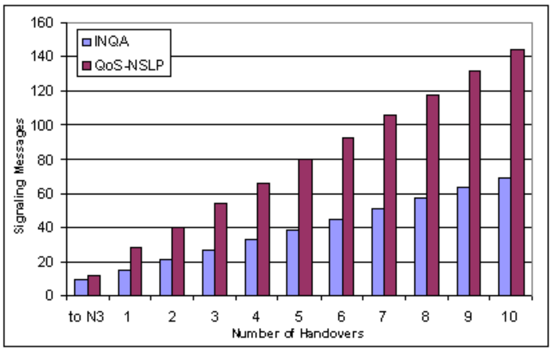
Using QoS-NSLP, N1 requests, before the handover, resources by sending a reserve message towards N3, via N2. After receiving the Response message, the application in N1 starts generating traffic. After the handover to N4, N1 reserves resources for the FTP flow in the new path towards N3, after being triggered by the connectivity module of the test-bed.

*1) Signaling Overhead:* To evaluate the signaling overhead, we measure the number of QoS-NSLP and INQA messages during the ping-pong movement scenario. Our findings are depicted in Figure 7. This figure shows the number of messages for two handovers (Figure 7a)), and the overall number of messages after 10 handovers (Figure 7b)). In case of two handovers, one to N3 and one to N4, we consider that there is already an SLS between N3 and N4. In this case, INQA needs six messages after each handover: three messages between the access networks (N3 or N4) and N2, and another three messages between N2 and N1. In case of the QoS-NSLP, Figure 7 a) shows that four messages are needed to set resources after the handover to N3, and six after the handover to N4. This increase is due to the longer path from N1 to N3, after the handover to N4. Although INQA requires more signaling messages than QoS-NSLP after the handover to N3, INQA messages are smaller than QoS-NSLP ones.

In a scenario in which N2 performs a ping-pong movement between N3 and N4 (Figure 7 b)), we consider that N2 spends two minutes within the range of N3 or N4 (i.e. one minute before it reaches N3 or N4 access points and one more minute

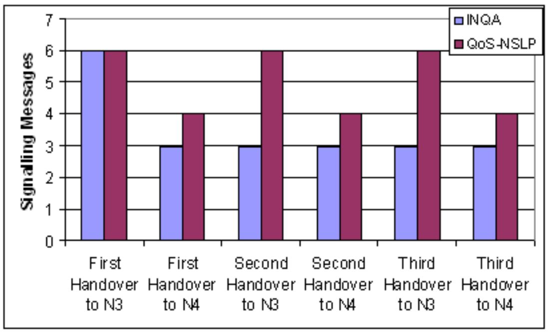


(a) One Handover

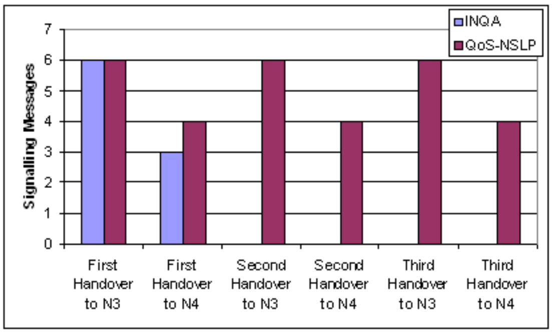


(b) Ping-pong movement

Fig. 7. Signaling overhead during movement



(a) First optimization level



(b) Second optimization level

Fig. 8. Optimized signaling overhead during movement

after it has stopped in the station) and stops for another two minutes when it attaches to N3 or N4. In this scenario we assume that there is no initial agreement between N3 and N4. Therefore, INQA sends nine messages at the beginning of this experiment: three messages between the stations (N3 and N4) and another six to establish an SLS between N3, N2 and N1. For any handover INQA needs six messages, as explained before.

On the contrary, the first time it comes into range of N3, QoS-NSLP needs four messages to reserve resources on the path and eight refreshing messages: two messages before it stops (one minute), four messages while stopped at the station (two minutes) and another two messages before the connection is lost (one minute). Hence, QoS-NSLP generates 12 messages after each handover to N3, being this number higher after the handover to N4, due to the longer path from N1 to N3 via N4.

In summary, as shown in Figure 7 INQA induces approximately 50% less signaling overhead than QoS-NSLP in a scenario with ping-pong movement.

*2) Signaling Overhead Optimization:* The previous section describes a worse-case scenario, since INQA can be more efficiently configured in a scenario with ping-pong movement between pre-determined access networks. This optimization can be done in two levels as illustrated in Figure 8. A first level encompassing a signal reduction between N2 and N1, and a second level encompassing a further signal reduction, between N2 and the train-stations N3 or N4.

In the first level of optimization, illustrated in Figure 8 a), INQA exchanges six messages after the first handover to N3 in order to set SLSs between N3 and N2 and between N2 and N1. In any subsequent handover INQA exchanges only three messages between N2 and N3 or between N2 and N4. This occurs since the SLS between N2 and N1 can be used after any handover.

In the second level of optimization, illustrated in Figure 8 b), INQA exchanges the same number of messages (as with the first level of optimization) between N2 and the train-stations after the first two handovers (six for N3 and three for N4). However, thereupon there are no more signal exchanges, since the previously negotiated SLSs can all be re-used. That is, N3 and N4 activate the previously negotiated SLSs each time they detect the attachment of N2 to their network. Keeping the SLSs when N2 is not attached does not lead to network resource wastage in N3 and N4, since resources are only allocated to an SLS during its *active-periods*. In a real scenario, the active-periods of an SLS, negotiated between N2 and each train-station, may be defined based on the train schedule.

*3) Communication Interruption Time:* In this section we analyse the interruption time of the communication during a handover, being the interruption time equal to the handover time (which is independent of the signaling approach), plus the convergence time of each signaling approach after the handover. Hence, we investigate QoS-NSLP and INQA convergence time after a handover from N3 to N4.

Figure 7a) shows that QoS-NSLP and INQA need six messages before they reach an equilibrium (i.e. generate traffic to N4). However, they have different signaling range: while INQA signals between N1, N2 and N4, QoS-NSLP signals among the entire end-to-end path (N1, N2, N4 *and* N3), making it dependent on the path length. Consequently, we expect INQA to converge faster than QoS-NSLP after each handover, due to the lower number of networks that have to be signalled, which reduces the round-trip-time.

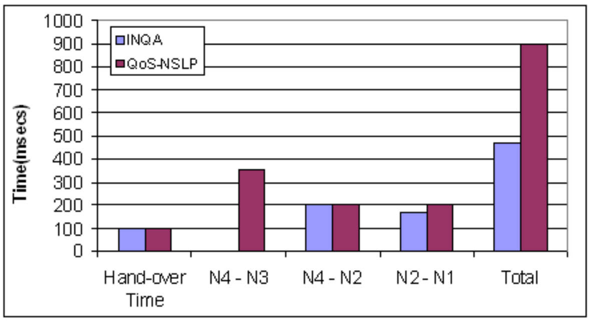


Fig. 9. Handover interruption times

Our assumptions match well our experimental results, depicted in Figure 9. This figure illustrates the handover time and the average converge times (during the five handovers performed to N4 in the ping-pong movement experiment) for each inter-network connection. Figure 9 shows that at the end of the handover, INQA needs approximately 180 to 200 ms to signal between any two networks, which means that INQA convergence time does not exceed 400 ms. This corresponds to the worse-case scenario, in which INQA needs to signal between N1, N2 and N4. On the other hand, QoS-NSLP needs to permanently signal on the complete path from the sender to the receiver, which means from N1 to N3, passing by N2 and N4. In our experiments QoS-NSLP needed around 800 ms to converge, after the handover to N4.

Since INQA only signals among local networks, such as a user, a train and an access-network, it takes advantage of the lower distance between them. On the contrary, QoS-NSLP behavior is subjected to the number of networks in the path to the receiver network. This is verified in the experiment results, which show that QoS-NSLP needs approximately the same time as INQA for the local signaling, but needs 300 ms more to signal from the access-network to the provider-network.

The different time needed by INQA and QoS-NSLP to signal between N1 and N2 occurs because the processing delay within each network oscillates between each of the five experiments.

This analysis refers to a handover from N3 to N4 during a ping-pong movement. In the reverse handover, the difference between the convergence time of INQA and QoS-NSLP lies in the order of 150 ms, on the favor of INQA. This is lower than the difference shown in Figure 9 (more than 500 ms), because after the handover to N3, QoS-NSLP needs to signal over three networks instead of four. As might be expected, in any experiment, packet loss rates are proportional to QoS-NSLP and INQA convergence time.

## VI. CONCLUSIONS AND FUTURE WORK

We analyzed the behavior of the flow-based and SLS-based signaling approaches, and concluded that the latter approach may bring more benefits for the next generation IP networks. Therefore, we propose a SLS-based signaling protocol, called INQA, to control inter-network QoS. A detailed experimental analysis is carried out with focus on the scalability and convergence time of QoS-NSLP (flow-based signaling approach) and INQA in a scenario with both stationary and mobile networks.

In the scenario with stationary networks, we observed that INQA introduces 40% less signaling overhead than QoS-NSLP when new networks attach to the chain of the already existing ones. The same overhead difference applies with the duration of communications. However, it is clear that INQA may have scalability problems with an increasing number of available services, something that calls for further investigation and requires the development of an aggregation mechanism, which constitutes subject of future work.

In the scenario with moving networks we observed that INQA induces 50% less signaling overhead than QoS-NSLP, in a scenario with ping-pong movement (this value corresponds to a worse-case scenario for INQA). In what concerns the convergence time, INQA converges faster than QoS-NSLP, by approximately 150 ms and 500 ms, depending on the path length (INQA advantage is proportional to the path length).

As future work, we will also analyse the trade-off between the refreshing mechanism of QoS-NSLP and the connectivity awareness of INQA to react to routing changes, in scenarios involving more frequent handover events.

## REFERENCES

[1] M. Boucadair. QoS-Enhanced Border Gateway Protocol. Internet Draft - Work in progress, Internet Engineering Task Force, July 2005.

[2] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. "Resource ReSerVation protocol (RSVP) – version 1 functional specification". RFC 2205, Internet Engineering Task Force, September 1997.

[3] W. Chao, J.P. Macker, and J.W. Winston. NRL Mobile Network Emulator. Technical Report, Naval Research Laboratory, January 2003.

[4] P. Chimento and B. Teitelbaum. Simple Interdomain Bandwidth Broker Signaling. Internet2 - Work in progress, Internet2, January 2000.

[5] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol. RFC 3626, Internet Engineering Task Force, October 2003.

[6] J. Manner, G. Karagiannis, and A. McDonald. NSLP for Quality-of-Service signalling. Internet Draft - Work in progress, Internet Engineering Task Force, January 2006.

[7] N. Niebert, A. Schieder, H. Abramowicz, G. Malmgren, J. Sachs, U. Horn, C. Prehofer, and H. Karl. Ambient Networks: An Architecture for Communication Networks Beyond 3G. *IEEE Wireless Communications*, 11(2):14–22, April 2004.

[8] Y. Rekhter and T. Li. A Border Gateway Protocol 4 (BGP-4). RFC 1771, Internet Engineering Task Force, March 1995.