The 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'07)

ANALYSIS OF METHODS FOR CONTROLLING QOS AGREEMENTS AMONG IP MOBILE NETWORKS

Eleni Kamateri and Lefteris Mamatas
Demokritos University of Thrace
Vas. Sofias 12, 67100 Xanthi, Greece
E-mail: {ekamater, emamatas}@ee.duth.gr

Paulo Mendes and Luis Loyola
DoCoMo Euro-Labs
Landsbergerstr 312, 80687 Munich, Germany
E-mail: {mendes, loyola}@docomolab-euro.com

ABSTRACT

Future mobile services are expected to have different levels of QoS requirements, raising the need to enhance networks with the capacity to differentiate among different classes of traffic, which may use different QoS models. To support and shelter different QoS models, future mobile networks should make use of an inter-network mechanism allowing the establishment of bi-lateral Service Level Specifications (SLSs), without making any assumption about the signalling supported by internal network devices. Hence, this paper aims to explore the best way to use bi-lateral signalling to build end-to-end chains of SLSs. Two methods are evaluated. In the first one, control messages are only propagated after the establishment of an SLS between a pair of networks; in the second one control messages flow from the provider of the SLS to each of its customers, passing a chain of transit-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 different available solutions to support or control different classes of traffic within the boundaries of each network, such as Multiprotocol Label Switching (MPLS) [1], the Integrated Services model (IntServ) [2] and the Differentiated Services model (DiffServ) [3]. Overcoming this heterogeneity is the first challenge for the control of end-to-end QoS. The described models assume that the same QoS model is used by all networks in the end-to-end path, which may not be true.

One initiative to overcome this limitation was given by the Next Steps in Signalling (NSIS) working group of the Internet Engineering Task Force (IETF), with the definition of the QoS NSIS Signalling Layer Protocol (QoS-NSLP) [4]. QoS-NSLP aims at reusing RSVP [5], while simplifying it, and adopting a more general signalling model. One of these generalizations is the definition of a QoS template that allows stacking different QoS specifications for messages crossing heterogeneous networks. Nevertheless, QoS-NSLP still requires all networks to support its state-machine in, at least, a subset of network devices in the data path. Besides, since it is based on per-flow signalling, it requires a significant overhead when terminals move, since a new end-to-end reservation must be set when a handover occurs.

To comply with the need to support and shelter different QoS models, future mobile networks should make use of an inter-network mechanism which allows the establishment of bi-lateral Service Level Specifications (SLSs), without making any assumption about the signalling-type supported by internal routers. One could argue that QoS-NSLP might support such inter-network mechanism if used between any two QoS controllers placed in adjacent networks, forming a peer-to-peer signalling relationship. Hence, any signalling process would start by a peer initiator sending a *Reserve* message to ask a downstream-peer to set an SLS based on the estimated QoS requirements of the initiator. Alternatively, the initiator-peer could send a *Query* message asking the downstream-peer about its capability to support specific QoS assurances.

While the QoS-NSLP *Reserve* process may lead to a try-and-error approach, the *Query* process may be too slow to react to the mobility of hosts[1]. A more suitable approach to handle bi-lateral agreements between adjacent networks may follow the process already used by inter-network routing protocols, such as BGP [6]. That is, QoS controllers may announce to their neighbours the capacity assuring a certain level of QoS to a certain type of traffic. Peer controllers, listening to these announcements, may decide to negotiate QoS guarantees to all or a subset of traffic types. Since this message sequence follows closely the advertisement process used by BGP, it may allow an easier interaction between the inter-network SLSs control mechanism and inter-network routing protocols.

This new advertisement/negotiation message sequence could be implemented by extending the current QoS-NSLP proposal. In that case, certain QoS-NSLP capabilities (such as signalling over a set of hops between an initiator and a receiver) are not needed.

Based on this analysis, the Ambient Networks project [7] developed a new solution for the dynamic control of inter-network QoS agreements, based on the advertisement and bi-lateral negotiation of SLSs. This SLS-based-signalling controls dynamic SLSs between networks and supports inter-network traffic engineering. Furthermore, the proposed solution may complement flow-based signalling approaches, such as in the following two examples: (i) the use of a reservation protocol for the immediate reservation of resources negotiated in SLSs; (ii) the use of a query protocol that checks the availability of resources in established chains of SLSs.

This paper aims to explore the bi-lateral signalling mechanism of InterNetwork QoS Agreements Protocol (INQA-P) [8], [9] that builds end-to-end chains of SLSs. Two hypothesis are evaluated: the basic protocol's behaviour, we call it INQA, in which control messages are only propagated after the establish-

---

[1] QoS-NSLP does not support network mobility

1-4244-1144-0/07/$25.00©2007 IEEE

ment of an SLS between a pair of networks. In this case, that a network only re-advertises an SLS after having negotiated it with its provider. An alternative behaviour, namely INQA-VAR, controls the messaging flow from a provider to each of its customer, passing by a chain of transit-networks. Although both approaches use bi-lateral communication, the QoS agreements in INQA take place between two neighbour networks (i.e., a bi-lateral agreement) and in INQA-VAR between two edge networks (i.e., an end-to-end agreement).

The remainder of this paper is organized as follows. In section II, we briefly analyse the advantages and disadvantages of the two signalling approaches. In section III, we discuss our evaluation plan, experimental scenarios and present our experimental results. Finally, in section IV we conclude our paper and enumerate some open issues.

II. SIGNALLING METHODS

According to INQA-P, a network can have one of the three following roles: a provider that advertises SLSs to other networks, a customer that negotiates SLSs or a customer-provider that resells SLSs advertised by its neighbours.

The INQA-P protocol maintains SLS state in adjacent networks, without the need of refreshing messages. Actually, each SLS carries an expiration time. Consequently, the state is kept until the SLS is expired. To control the SLS state, INQA-P uses four message types: *Advertisement*, *Negotiation*, *Acknowledgement*, and *Monitoring*. The *Advertisement-message* is used as an announcement for unexploited SLSs to a set of neighbour networks. The *Negotiation message* is used by a customer or a customer-provider to reserve previously advertised resources. After a successful negotiation (initiated by an *Acknowledgement-message* send by the provider), new SLS state is allocated. In untrusted environments, a customer-network may query (with a *Monitoring-message*) the provider-network about the level of the provided service in order to check its consistency to the negotiated SLS. We present two different ways of bi-lateral signalling that build end-to-end chains of SLSs.

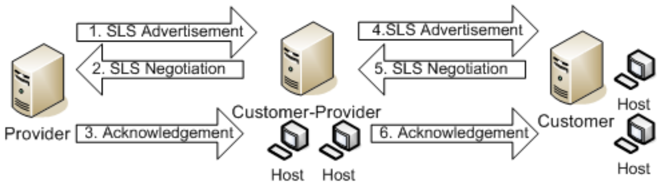

Figure 1: Message-Sequence of INQA Signalling Method

A. Signalling for bi-lateral QoS agreements (INQA)

In Figure 1, we show the sequence of signalling messages for bi-lateral QoS agreements (INQA). In INQA, a provider-network offers its services by sending an *Advertisement-message* (message 1) to its adjacent customer-provider network. The latter requests the offering resources via a *Negotiation message* (message 2). The provider acknowledges the reserved resources to the customer-provider (message 3). Hence, the customer-provider network may either utilize all the negotiated resources or utilize only part of them and sell the rest to the next customer-network or customer-provider network. The same process may repeat (e.g., messages 4, 5 and 6) until the resources are exhausted. This chain of bi-lateral agreements allows a service to be offered end-to-end.

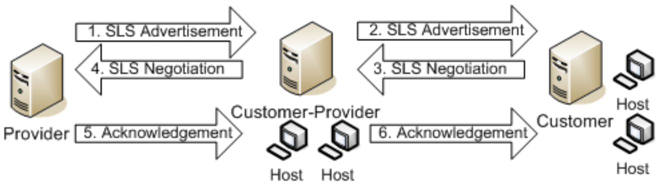

Figure 2: Message-Sequence of INQA-VAR Signalling Method

B. Signalling for end-to-end QoS agreements (INQA-VAR)

In Figure 2, we show an alternative sequence of messages for an end-to-end QoS agreement (INQA-VAR). In this case, a provider offers its services by sending an *Advertisement-message* (message 1). In INQA-VAR, a customer-provider network acts as a transit-network that relays any SLS sent to it (message 2). Actually, a customer-network initiates a distant agreement with the provider, negotiating the offered services through a chain of transit-networks (messages 3, 4). The provider-network acknowledges the agreement with the customer-network via an *Acknowledgement-message* that crosses the same chain of transit-networks (messages 5, 6).

C. Discussion

We note that although both methods differ in the re-selling policy, they use the same number of messages for the establishment of a QoS agreement (Figures 1,2). In INQA, every intermediate customer-provider should first negotiate and afterwards resell the offered resources while in INQA-VAR they act as purely transit-networks that forward every message they receive.

In INQA, *Advertisement-messages* may not reach all customer-networks because the resources can be exhausted by customers that are located closer to the provider. In contrast to INQA, INQA-VAR distributes *Advertisement-messages* to every customer-network.

III. EXPERIMENTAL EVALUATION

In this section, we detail our experimental evaluation. We discuss the topology used, the performance metrics and other specific parameter adjustments (such as the parameterization of SLSs and negotiation-profiles). We conclude this section with our experimental results.

A. Evaluation Details

We carried out simulations in NS-2 [12] that evaluate the scalability properties of the two signalling methods. We used a linear topology with 10 networks in the backbone line: one provider-network and one customer-network at each edge of the backbone, and eight customer-provider networks between the former two. As illustrated in Figure 3, a variable number of customer-networks (from 1 to 10) were connected to each customer-provider network, depending on the details of each particular experiment. Furthermore, each customer-network consists of ten local hosts which are the end-users of the traffic assurances. Each end-host runs three different applications (see Figure 3, where *3 appl* represent the three assigned application to each end-host)

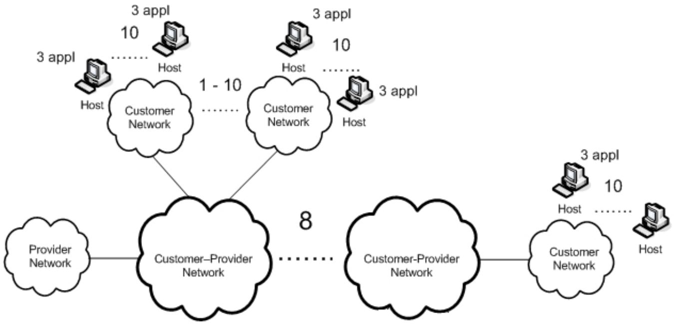

Figure 3: Experimental Topology

For each experiment, we measured the total transmitted data (in number of messages and bytes), the convergence time[2] (in seconds), the memory-state (in number of stored messages), and the satisfied applications ratio. The ratio of the number of matched negotiation profiles to the total number of customer-assigned negotiation profiles, both matched and unmatched.

All of our experiments have a non-deterministic nature. Consequently, we executed each experiment 20 times using different initiation values. Each of our Figures depicts average values of each metric.

B. *Adjustments of QoS-related Parameters*

In our experiments, only the Provider-Network advertises resources. SLSs are assigned to the Provider-Network in the beginning of the experiment.

We assume that each customer-host runs three applications. Practically, we set three different random-generated negotiation-profiles (one for each application) to each customer-host. The bandwidth parameter of the negotiation-profiles is set based on the bandwidth requirements analyzed in [10].

Each customer-provider, since it is connected with 10 customer-networks (see Figure 3), defines the type of SLS offers that is willing to accept, reflecting the needs of its local customer-networks and thus their customer-hosts. The bandwidth associated with each customer-provider's negotiation-profile is obtained by summing the bandwidth of similar negotiation-profiles/classes of its customer-networks.

In case of INQA, a customer-provider network accepts any offer (advertised SLS) that matches[3] one of its negotiation-profiles. Although the offered SLS may not match any of its negotiation-profiles, the customer-provider network will always accept the SLS, with the intention of "re-selling" it to other customer-provider networks.

We assume that the assigned bandwidth to the provider is randomly estimated using a uniform distribution between the two boundaries shown below:

$$MinBandwidth = maxTR - 10\frac{maxTR - minTR}{100} \quad (1)$$

$$MaxBandwidth = 2 * maxTR \quad (2)$$

where *TR* represents the total required bandwidth[4] for a specific application. Since a provider should offer sufficient resources to its customers, each advertised SLS covers the unique requirements of each application. In order to have unsatisfied applications in our results, we set the value of the minimum bandwidth (*MinBandwidth*) to a value that is 10% lower than the total rate (*TR*).

In [11], the NSIS Working Group recommends a classification of all applications into 8 network classes, according to their special demands in terms of Delay, Jitter and Packet Losses. We use the same classes (except class 5 that is related to the best-effort service) for any generated SLS. Each SLS is defined in terms of delay, loss, jitter and bandwidth. We computed the delay, jitter, and packet loss that are sold by each network to the next one (in the topology chain shown in figure 3) as follows:

$$D_P = \frac{D_T}{m+1} \quad (3)$$

$$J_P = \frac{J_T}{m+1} \quad (4)$$

$$L_P = 1 - \sqrt[m+1]{1 - L_T} \quad (5)$$

where *m* represents the position of the customer-provider networks in the backbone line (from left to right) in Figure 3 (starting from m=1), and D_T, J_T, L_T are the maximum tolerable end-to-end delay, jitter and loss for a specific class, respectively (following the traffic specification of [11]).

Furthermore, we introduced an 1% probability of having insufficient resources in order to produce results with unsatisfied customers.

C. *Experimental Results*

We evaluated the scalability properties of the two signalling methods using two different scenarios. In the first scenario, we evaluated INQA and INQA-VAR when the number of assigned customer-networks increases. We adjusted the number of customer-networks from 1 to 10 and kept the number of SLSs fixed (equal to 10). In the second scenario, we explored the impact of the number of SLSs in the system by attaching a fixed number of 10 customer-networks to each customer-provider and increasing the number of SLSs from 1 to 10.

---

[2] it is the elapsed time until all negotiations are completed and the system reaches a stable state

[3] we have a match when the offered bandwidth defined in the SLS is equal or higher than the one defined in the negotiation-profile

[4] for all customer-networks and thus customer-hosts

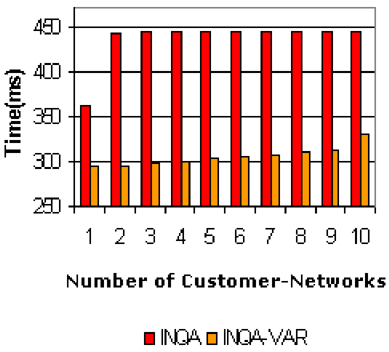

Figure 4: Convergence Time as a function of The Number of Customer-Networks

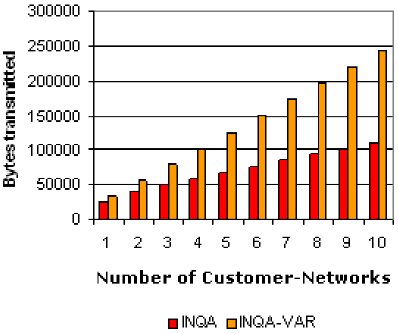

Figure 5: Communication Overhead as a function of Number of Customer-Networks

As we can see in Figure 4, INQA-VAR signalling method converges more than 3 times faster than INQA. This is because the intermediate customer-providers operate as transit-networks that forward any SLS offer without additional checks, since they do not have any assigned negotiation-profiles. In contrast, INQA's intermediate customer-providers spend extra time checking each SLS and iterate through all negotiation-profiles until a match is found.

The number of customer-networks impacts on the convergence time of the INQA-VAR method slightly (Figure 4). More precisely, the system's convergence time increases by 0,74% for each new customer-network, whereas in INQA it is almost stable (for more than one customer-networks). In INQA-VAR, if a negotiation-profile is matched, a QoS agreement between a customer and the provider will set up. As the number of the QoS agreements increases, the number of negotiation queries stored in the provider's database is reflected accordingly. On the contrary, in INQA signalling method, *Negotiation messages* are processed by each customer-provider in parallel (therefore no buffering is involved). Hence, when the number of customer-networks is more than one, INQA has a fixed convergence time (Figure 4).

We observe in Figure 5 that INQA method transmits fewer messages (and thus bytes) than INQA-VAR (i.e., 44,44% less bytes). The end-to-end method of INQA-VAR introduces extra communication overhead because all customers need to establish an agreement with a single provider. Consequently, their messages need to traverse a number of hops. In contrast, the bi-lateral method of INQA is based on local agreements between two neighbouring networks. We conclude that INQA introduces less communication overhead than INQA-VAR but converges later.

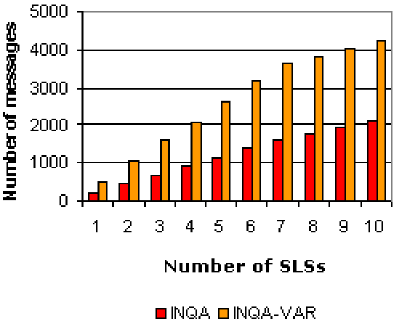

Figure 6: Memory Consumption as a function of The Number of SLSs

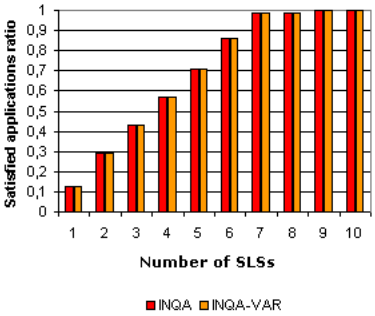

Figure 7: Satisfied Applications Ratio as a function of The Number of SLSs

While INQA signalling method establishes an agreement between a provider and a next-hop customer, in INQA-VAR the same agreement is taking place between a provider and a customer regardless of its location in the topology. The transit-networks situated between the provider and the customer (negotiating for the offered resources) assist to the maintenance of this agreement by keeping it in their databases. In INQA-VAR, the overall number of messages kept in the databases is related to the distance (in hops) between the customers negotiating for the resources and the provider. For the same ratio of satisfied applications, INQA stores almost half the number of messages in memory (Figure 6). Consequently, INQA appears more efficient in terms of memory consumption (55,05% less memory consumption).

As we can see in Figure 7, the two protocols satisfy the same number of applications. Practically, the two methods end up in the same stable state, where resources are allocated to each node in the same manner. We note that they do not achieve a

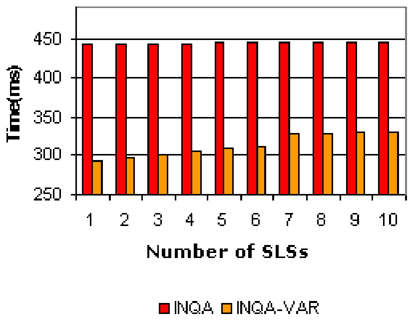

Figure 8: Convergence Time as a function of The Number of SLSs

rate of 100% satisfied applications, for 7 and 8 SLSs, because we used a uniform distribution in the assignment of the QoS requirements of the networks. This introduced a probability 1% of insufficient delay, 1% for insufficient jitter, 1% for insufficient loss and 10% for insufficient bandwidth. Consequently, a few unsatisfied applications can be noticed in Figure 7.

In Figure 8, we show that the smaller convergence time of INQA-VAR prevails even with an increasing number of SLSs (i.e., converges almost 3 times faster than INQA). Although the number of SLSs significantly impacts on the convergence time of INQA-VAR (i.e., an 1.34% increase for each new SLS), INQA needs a fixed amount of time to converge (i.e., 446msec for any number of SLSs). When the QoS controller releases more than 7 SLSs, the redundant SLSs (8th, 9th, 10th) are a repetition of the previous ones (specifically 1st, 2nd and 3rd), because the classification of traffic is up to 7 classes. In case of more than 7 SLSs, the convergence time of INQA-VAR appears stable. For example the case of 10 SLSs, the last three SLSs are going to be negotiated only by a customer which remains unsatisfied with the first three SLSs.

IV. CONCLUSIONS

In this paper, we analysed and evaluated two alternative methods to use bi-lateral signalling to build end-to-end chains of SLSs: (i) INQA, a method in which control messages are only propagated between a pair of networks after the establishment of an SLS; (ii)INQA-VAR, where control messages flow from the provider of the SLS to all customers, passing a chain of transit-networks. We carried out the experiments and concluded that INQA method brings more benefits in terms of memory consumption and communication overhead than INQA-VAR. However, we observed that the bi-lateral signalling method introduces significant latency to the system due to its effort for a stable state, contrary to the end-to-end method.

The above observations call for new issues that need to be tackled. As future work, we plan to focus on the above trade-off between the convergence time and the signalling overhead, so that INQA-VAR could be more suitable for data traffic and INQA for real time traffic.

REFERENCES

[1] E.Rosen, A. Viswanathan, R. Callon, et al., ”‘Multiprotocol Label Switching Architecture.”’ *IETF Internet Draft*, Aug. 1999.

[2] R. Braden, D. Clark, and S. Shenker, ”‘Integrated services in the internet architecture: an overview.”’ *RFC 1633*, June 1994.

[3] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, ”‘An architecture for differentiated services.”’ *RFC 2475*, December 1998.

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

[5] 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.

[6] Y. Rekhter and T. Li. ”‘A Border Gateway Protocol 4 (BGP-4).”’ *RFC 1771*, Internet Engineering Task Force, March 1995.

[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):1422, April 2004.

[8] P. Mendes, J.-A. Colas, C. Pinho ”Information Model for the Specification of QoS Agreements among Ambient Networks.”, *IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, Berlin, Germany, September 2005.

[9] I. Psaras, L. Mamatas and P. Mendes ”QoS Control in Next Generation IP Networks: An Experimental Analysis of Flow-based and SLS-based Mechanisms.” *In Proceedings of the Workshop on Networking in Public Transport (WNEPT 2006)*, Ontario, Canada, August 2006.

[10] A.-G. Acx, J. Henriksson, B. Hunt, G. T. Karetsos, E. Tragos, A. Mihovska, S. Kyriazakos, L. Moretti, J. Lara. ”‘IST-2003-507581 WINNER D1.3 version 1.0”’, Final usage scenarios 30/06/2005

[11] J. Ash, M. Dolly, C. Dvorak, A. Morton, P. Taraporte, and Y. E. Mghazli. ”‘Y.1541-QOSM - Y.1541 QoS Model for Networks Using Y.1541 QoS Classes.”’ *Internet Draft, Work in progress,* NSIS Working Group, May 2006.

[12] The Network Simulator - NS-2, Web Page: http://www.isi.edu/nsnam/ns/