Signaling Environment for Service Level Negotiation and QoS management in Mobile IP Networks

Badr Benmammar, Nader Mbarek, Francine Krief

Signaling Environment for Service Level Negotiation and QoS management in Mobile IP Networks

Badr Benmammar, Nader Mbarek and Francine Krief
LaBRI Laboratory - CNRS UMR 5800
University of Bordeaux I
Talence 33400, France
{benmamma, mbarek, krief}@labri.fr

Abstract

This paper describes a signaling environment for service level negotiation and advance resource reservation in mobile IP networks. This environment is built in conformance with the generic signaling environment, which is standardized by the NSIS IETF working group. The service level negotiation protocol, called SLN NSLP, integrates the NSIS architecture to assure, thanks to the negotiation of an SLS, the availability of the resources needed by applications. The advance resource reservation protocol, called MQoS NSLP, is based on the QoS NSLP signaling application. It provides to mobile terminals the QoS required by taken into account the user's mobility and QoS profile. The advance resource reservation is based on an object MSpec (Mobility Specification) which determines the future location of the mobile terminal. The MSpec object which is a part of a mobility and QoS profile, is determined by the mobile terminal.

1. Introduction

In 2002, the IETF has launched the Next Steps In Signaling working group (NSIS). The initial objective of this group was to unify all the existing solutions of IP signaling or to make them coexist. With the emergence of IP networks and the increasing number of applications requiring a high level of QoS, the signaling problem became increasingly critical. Provide a universal signaling, which takes into account the QoS as well as the security and the mobility is a very difficult task. Initially, the NSIS working group aimed the QoS, and proposed the QoS NSLP [1] signaling application.

The objective of our work is to propose a signaling environment for service level negotiation and advance resource reservation in mobile IP networks in conformance with the generic signaling environment standardized by the NSIS IETF working group. First, we propose a service level negotiation protocol that we call SLN NSLP (Service Level Negotiation NSLP). This protocol, in collaboration with the other layers of the NSIS architecture, should permit to guarantee the availability of a certain level of service following the negotiated SLS. Then, we propose to use the QoS NSLP messages in order to make resources reservation in advance in order to reduce the impact of the handover on the quality of service. This reservation is based on an object called MSpec (Mobility Specification) that determines the future locations of the mobile terminal. The MSpec object is a part of a mobility and QoS profile, which is determined by the mobile terminal. We propose a format for this object, which will be included in the QoS NSLP messages.

This paper is organized in five sections. The first section presents a synthesis of the research relating to resources reservation in an IP mobile environment. The second section discusses the interest of a SLS negotiation. The third section defines the NSIS environment in which we specify the SLS negotiation protocol and the advance resource reservation protocol. Then, we propose a mobility and QoS profile and describe the format of the MSpec object. The last section is a description of a case of utilization of this signaling environment.

2. QoS in Mobile IP Networks

One of the major challenges in mobile environments is the provision of Quality of service (QoS) guarantees that different applications require considering the highly dynamic nature of these environments. Recent research takes an interest in advance resource reservation to provide the necessary QoS to the mobile terminals. In the integrated services networks, the majority of research is interested in extending the RSVP protocol in a mobile environment.

The authors in [2] proposed a new protocol of resource reservation in mobile environment called MRSVP. In this model of reservation, the mobile terminal can make advance reservations in a set of cells named MSPEC (Mobility Specification). The MSPEC is not very clear, it only indicates the future locations of the mobile terminal but the MSPEC is not described. Authors proposed other RSVP messages in order to treat the user's mobility. This technique requires additional classes of service, major changes of RSVP, and a lot of signaling.

The same authors in [3, 4] described an architecture, which supports *mobility independent* and *dependent services* in a same network. In this architecture, the concept of *active* and *passive* reservation is used to obtain a better use of resources. The reservation for a flow in a link is called active, if the packets of this flow pass through this link in order to reach the receiver. The reservation is called passive, if the resources are reserved for this flow on the link, but the current packets for this flow are not transmitted on this link. The resources of the passive reservation can be used by other flows, which do not require QoS guarantee, like BE (Best Effort) flows.

Min-Sun Kim and al [5] proposed a resource reservation protocol in a mobile environment. The proposed protocol introduces the *RSVP agent* concept in order to guarantee the necessary QoS through an anticipation of the resource reservation. In this protocol, there are 3 classes of resource reservation to obtain a better use of resources:

- *The Free class:* it represents the resources used in Best Effort.
- *The Reserved class:* it represents the reserved resources for a specific flow, which are currently used.
- *The Prepared class:* it represents the reserved resources for a specific flow, which are not currently used.

3. Service Negotiation

The increasing need of mobility requires the dynamic negotiation of service level between users and service providers. Therefore, protocols like COPS-SLS [6], DSNP (Dynamic Service Negotiation Protocol), SrNP (Service Negotiation Protocol) [7], SLS IPCP (Internet Protocol Control Protocol) or projects like Mescal (Management of End-to-end Quality of Service Across the Internet at Large) [8] and Cadenus (Creation And Deployment of End-User Services in Premium IP networks) [9] contributed to the dynamic aspect of the service level negotiation. The SLS parameters, which are dynamically negotiated, are defined in the Tequila project [10].

To define the requirements of a service level negotiation protocol that we want compliant to the NSIS architecture, we sequenced these requirements in two categories: those that ensue of the specificities of the procedure of negotiation and those that have been defined for every NSLP.

The first category is important to allow negotiation entities to initiate, modify or release a level of service already established following a previous negotiation. A Supplier can initiate a modification to promote new capacities or for a deterioration of the service level due to congestion problems for example. on the other hand a Customer needs can change so he asks for a modification of an old negotiated SLS. It is one of the most important needs for the service level negotiation protocol in order to make the SLS negotiation dynamic. Nevertheless, the dynamic aspect of the negotiation depends on the negotiation entities. In fact, a SLS negotiated between two network providers is less dynamic than a SLS established between an user and his service provider, this could be explained by the amount of modifications that will take place on the network to offer the level of service guaranteed by the SLS. This dynamic aspect can be managed by the interval of renegotiation parameter of the SLS. Furthermore, the service level negotiation protocol should be independent of the SLS parameters in order to facilitate its extension in the future by the definition of new parameters of QoS, mobility or security [11].

By defining a service level negotiation protocol in the NSIS environment, we have to be in conformance with the NSIS requirements specified in [12]. Therefore, this protocol, unlike the other negotiation protocols (COPS-SLS, SrNP) will be an in band signaling protocol i.e. the messages sent in the negotiation procedure will be forwarded on the data path by nodes that implement SLN NSLP. No restriction is made to define the place of the SLN NSLP nodes in the network.

4. Signaling Environment

The IETF decided in 2002 to launch the NSIS working group to try to unify all solutions of signaling or to make them coexist. This group standardized two-layer architecture: the NSLP layer to generate signaling flows for different purposes and the NTLP layer to transport those flows in a path coupled way. Some concepts used in this architecture are inspired from RSVP but a modification and a simplification were made to support generic signaling.

4.1 GIST

GIST (General Internet Signaling Transport) is the protocol that NSIS has adopted as a standard for the NTLP layer [13]. GIST is conceived for an in band transport of signaling flows generated by the NSLP layer, i.e. signaling flows follow the same path as the data flows. Besides, it only treats unicast signaling. Finally, GIST collaborates with the underlying transport and security layers to assure the good routing of signaling flows. Two functions are assured by GIST:

- Routing: to determine the next adjacent GIST node on the data path.
- Transport: GIST uses the datagram mode with UDP to discover GIST nodes on the data path thank to specific RAO (Router Alert Option) whereas the connection mode is used with TCP or SCTP to transport the NSLP flows.

4.2 QoS NSLP

Whereas the NTLP layer has for essential goal the transport of signaling, the NSLP layer assures the generation of this signaling in accordance with of user needs. QoS NSLP [1] is the first NSLP layer protocol to be elaborated in NSIS:

it permits to generate a signaling to provide a certain level of QoS by making reservations on the data path independently of the QoS models (Diffserv, Intserv...) adopted by the different domains. With NTLP, QoS NSLP spreads functionalities of RSVP, such as the creation, the refreshing, the modification and the elimination of a reservation state. On the other hand, QoS NSLP proposes an interaction with the RMF (Resource Management Function) for the access control in accordance with specified control policies.

NSLP leans on NTLP so that signaling generated by its applications is transported correctly toward target nodes, that's why QoS NSLP, an example of NSLP protocol, interact with the lower layer NTLP in order to achieve these objectives. NTLP is independent of the NSLP layer signaling application and it is through the intermediary of one API that parameters asked by one layer are obtained.

QoS NSLP generates 4 messages types:

- *Reserve*: the only message, which handles the reservation state (refresh, create, remove).
- *Response:* using this message, a response is sent to a message received.
- *Query:* this message is used to require information concerning the nodes, which are on the data path, for example: the available resources.
- *Notify*: using this message, it possible to inform a node without preliminary request.

All QoS NSLP messages contain a common header, followed by objects whose the use is given according to the type of each message. Among the objects we find: Response Request, Refresh Period, Session Id, Error Spec, QSpec.

We are interested particularly in the QSpec object, because this object is used to specify the desired QoS. The following parameters are proposed for the QSpec [14]: QSpec ID, QSM Control Information, QoS Description: Traffic Descriptors, QoS Class, QoS Characterization, Excess Treatment, Priority and Reliability, Service Schedule, Monitoring Requirements.

4.3 MQoS NSLP

We name MQoS NSLP, the procedure of resources reservation in advance using the QoS NSLP messages in a mobile environment. This procedure of reservation is applied in a HMIPv6 architecture. The MAP (Mobility Anchor Point) plays a significant role to reserve the resources in advance on behalf of the mobile terminal.

QoS NSLP operates according to the two following modes: *Sender Initiated Reservation* and *Receiver Initiated Reservation.* In the first mode, the sender of the flow initiates the reservation (he generates the RESERVE message). In the second mode, the reservation is initiated by the receiver of the flow.

The MH can be a sender or a receiver of the flow, so there are 4 possible scenarios:

- The MH is the receiver of the flow with the mode *Sender Initiated Reservation.*
- The MH is the receiver of the flow with the mode *Receiver Initiated Reservation.*
- The MH is the sender of the flow with the mode *Sender Initiated Reservation.*
- The MH is the sender of the flow with the mode *Receiver Initiated Reservation.*

The advance resource reservation is based on an object MSpec (Mobility Specification) which determines the future location of the mobile terminal. This object is defined in section 5. An example of advance resource reservation by using MQoS NSLP is given in section 6.

4.4 SLN NSLP

SLN NSLP is in conformance with the NSIS requirements. Therefore, the messages sent in the negotiation procedure will be forwarded on the data path by nodes that implement SLN NSLP.

We will use the following terminology:

- *SNE*: NSIS entity that supports SLN NSLP.
- *SNI*: SNE node that initiates the procedure of negotiation.
- *SNR*: the last SNE node on the data path, it answers the SLN NSLP messages received.
- *SNF*: SNE node between the SNI and the SNR, it forwards the SLN NSLP messages toward the following SNE.
- *Sender*: a node originating packets of the data flow which are the purpose of the negotiation, it can be also an SNI.
- *Receiver*: a node that is the destination of packets of the data flow, it is different of SNR if it is a host.

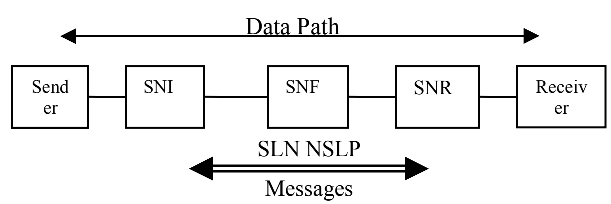


Figure 1. SLN NSLP entities

The figure 1 represents the SLN NSLP entities used in the negotiation. One or several SNF can be present on the data path. We notice that a SNE can be considered as a SNI, SNF or SNR according to the role that he plays in the procedure of negotiation. Besides, the negotiation is considered inter or intra domain according to whether the SNEs participating to the negotiation, are within the same domain or not.

SLN NSLP messages are sent peer-to-peer through the different SNE nodes present on the data path. This means that in an SNE node, when an SLN NSLP message is received, a new one (different or not) is generated and sent to the adjacent SNE.

We define for SLN NSLP six types of messages:

- ***Negotiate***: It is a message sent by the SNI toward the SNR. It permits to specify the SLS parameters we will negotiate as well as their values. The parameters are mentioned by flags in the header of the message and the values by TLV (Type Length Value) objects following the header. This message is interpreted by the adjacent SNF on the data path and another Negotiate message (modified or not) is sent toward the following SNF until the SNR.
- ***Revision***: It is sent by the SNR toward the SNI to propose an alternative to parameters and/or values received in the Negotiate message. This message can contain the SLS ID that is going to be used for the registration of the SLS if the negotiation will succeed.
- ***Response***: it is sent by the SNR or the SNI following a message containing a Response Request object in order to accept or reject a demand or a revision so that we finish the negotiation process reasonable delays. A Response message that accepts the negotiation must contain the SLS ID.
- ***Modify***: It is sent by the SNI toward the SNR with the SLS ID recorded after a successful negotiation. This modification can either be negotiated or an immediate answer can be asked by the SNI via the Response Request object and the procedure is finished by a Response from the SNR.
- ***Notify***: This message is sent by the SNR to ask the SNI to degrade the level of service that has already been negotiated. This message contains obligatory a Response Request object. This degradation is accompanied by penalties for the provider according to an SLA (Service Level Agreement).
- ***Release***: It is sent by the SNI toward the SNR to terminate a negotiated SLS. It contains the SLS ID. The corresponding information in the SLS repository will be suppressed in every SNE when it receives a positive Response.

All those messages have a common header with a two Bytes Message Type field and a two Bytes Flags field. The flags are used to indicate the presence of the TLV objects such as the SLS parameters objects or the Response Request object. A non-used flag has a zero value.

A TLV object comes after the header in the SLN NSLP message and it has the structure shown in the figure 2.

Length (16 bits)	Class-Num (8 bits)	C-Type (8 bits)
Object Value		

Figure 2. Structure of a TLV object

For a TLV object representing an SLS parameter that we will negotiate, the Class-Num field is identification for this parameter whereas the C-Type field specifies if the object values are quantitative or qualitative.

The FSM that we define for an SNE includes three states: E0 is the free state in which the entity didn't start the procedure of negotiation; E1 is the state during which an entity is waiting for a message from an other entity and finally the state E2 represents the end of the negotiation. The transition between the states is a result of a reception and/or the sending of an SLN NSLP message. It is out of the scope of the FSM that we define to consider the external interactions with the Decision Module, QoS NSLP and GIMPS, nor the internal interactions like the timers. The figure 3 represents the FSM of the SNI.

The SLS parameters are transported by the messages of SLN NSLP as a TLV objects and mapped after the end of the negotiation into the fields of the QSPEC which is transparent to QoS NSLP but specified by a QSP (QoS Signaling Policy) [14] specific to a QoS model (Diffserv, Intserv…). A template for the SLS parameters are specified in [10]. The most important parameters are: Service Schedule, Performance Guarantee (Bandwidth, Jitter, Delay, Loss Rate), Description and Traffic Conformance (Token Bucket), Excess Treatment, Negotiation Mode (predefined SLS or not) [6], Renegotiation Interval. The flags with a value of 1 in the header indicate the parameters that will be negotiated by SLN NSLP as shown in figure 4.

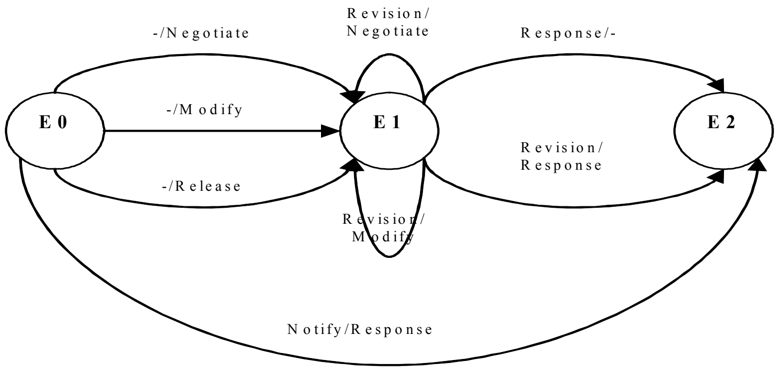


Figure 3. The SNI FSM

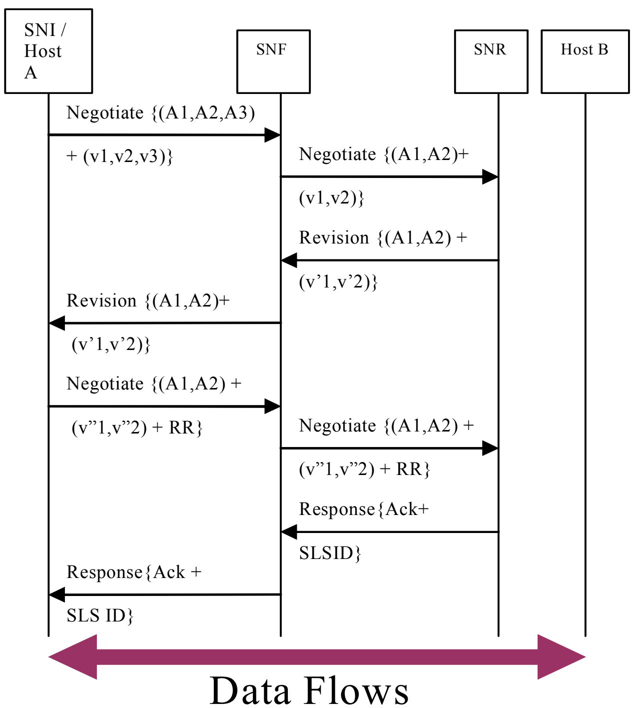


Figure 4. An example of simple negotiation with SLN NSLP

The SNI initiates the negotiation with an SLS containing the attributes (A1, A2, A3) with the values (V1, V2, V3) and finally the negotiation finished with a RR (Response Request) that results in the sending of Response message with an SLS ID corresponding to (A1, A2) with the values (V'1, V'2).

4.5 Interactions

The figure 5 shows the interactions between the different protocols and NSIS modules. When an SNE node receives a negotiation request, an interaction takes place with the local RMF to see if there are enough available resources thanks to the Admission Control module and if the demand has the necessary authorizations thanks to the Policy Control module. The decision that is going to be taken by a SNE to answer to an SLN NSLP message (acceptance, dismissal, alternative values...), or the modification or not of the message to forward, is influenced by the interaction with the RMF as well as a Decision module. In the SNI, This module is going to generate values that we are going to negotiate. Once the negotiation finished, a horizontal interaction will take place between SLN NSLP and MQoS NSLP. SLN NSLP will map the negotiated SLS parameters into the QSPEC [14] MQoS NSLP object so that QoS NSLP reserves the resources corresponding to the negotiated SLS. Besides, an SLS ID will be passed to MQoS NSLP so that the reservation corresponding to this negotiated SLS is automatically accepted without requesting the RMF and the Policy Control (this procedure [1] was obligatory for QoS NSLP nodes without SLN NSLP implementation). We make it possible thanks to the definition of the SLS repository where we register the characteristics of the SLS ID to obtain the parameters of the corresponding negotiated SLS when it is invoked for a reservation or a modification.

MQoS NSLP will reserve resource according to the SLS negotiated and the QoS profile. This profile which contains the Mspec object, is defined in the next section.

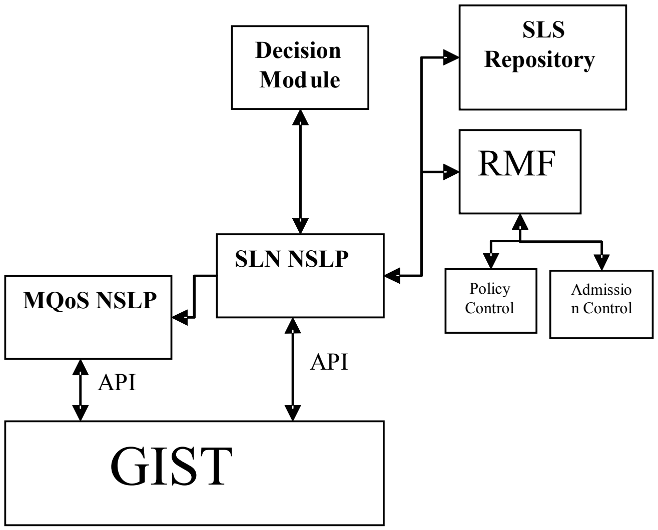


Figure 5. Signaling environment in an SNE node

5. Mobility and Qos Profile

The user's mobility profile is built on the basis of its behaviour / movement after **m** associations with the system. The goal of this profile is to build a user's behaviour model.

The system model is based on the Continuous Time Markov Chain (CTMC).

Our system can evolve between N states defined by the following set: $C = (C_1, C_2, \ldots\ldots C_i \ldots\ldots C_n)$.

The system is in the state i = the terminal mobile is in the cell C_i.

P_{ij}: the probability of transition from the cell C_i to the cell C_j.

$P_i\ (t_r)$: the probability, which defines the location of the mobile terminal in the cell C_i at the time t_r.

The user's mobility profile contains the following information:

- The user's identifier;

- User Preferences: User_P;

This attribute represents the set of the user's preferences.

The user's preferences are determined after an observation phase in which the system observes the user's behaviour.

The following diagram represents the determination of the User_P:

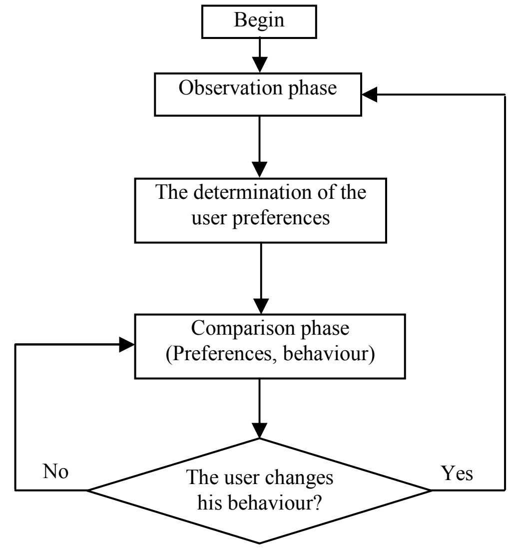


Figure 6. The determination of the User Preferences

The proposed format for the User_P is as follows:

User_P = <Preference ID> <Duration_P> <Cell_P> < QoS_level>

- <Preference ID>: it identifies the preference (the system can detect several preferences for the user).
- <Duration_P> : <start_P> <end_P>: it determines the period of time in which the user's preference is satisfied.
- <Cell_P>: it determines the cell in which the user's preference is satisfied.
- <QoS_level>: it is the QoS level needed by the user for the preference.

- M = $[P_{ij}]$ [N*N]: The Matrix of transition, which contains the P_{ij}, before the **m** associations, the P_{ij} are random.

We note:

t [i, j]: the number of transition from the cell i to the cell j during the **m** associations with the system.

g (i): the number of transition outgoing from the cell i during the **m** associations with the system. We calculate it as follows: $g(i) = \sum_{j=1}^{n} t[i, j]$.

After the **m** associations, the probability of transition from the cell i to the cell j is calculated as follows:
$P_{ij} = t[i, j] / g(i)$.

- $V = [P_i(t_o)]$ [N]: This Vector contains the $P_i(t_o)$.

We note:
$P_i(t_o)$: this probability defines the location of the mobile terminal in the cell C_i at the time t_o.
Before the **m** associations, the $P_i(t_o)$ are random and after each association with the system the time is initiated to t_o.
k (i): the number of association with the cell i during the **m** associations at the time t_o.

We have: $\sum_{i=1}^{n} k(i) = m$ and $P_i(t_o) = k(i) / m$.

The figure 7 represents the determination of the M Matrix and the V Vector:

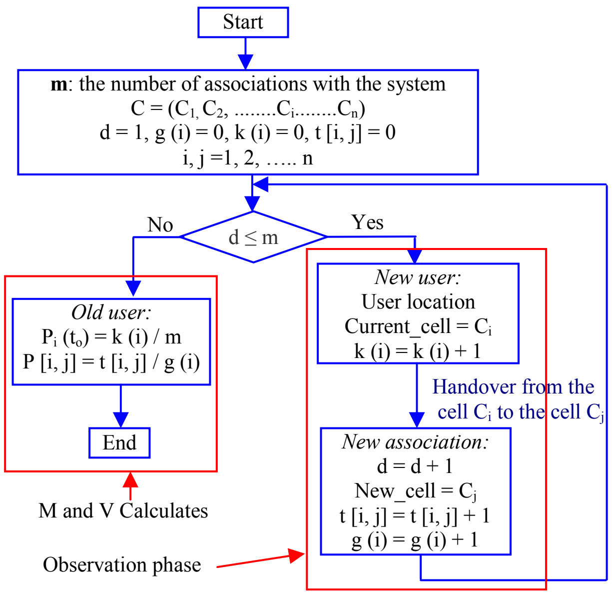


Figure 7. The determination of the M Matrix and the V Vector

- The MSpec (Mobility Specification): The MSpec determines the future locations of the mobile terminal.

The proposed format for the MSpec is as follows:
MSpec = <MSpec ID> <Duration> <Cell ID>.
- *MSpec ID* is the identifier of the MSpec.
- *Duration* is the interval of time (< start time>, <end time >) during which the future locations of the mobile terminal can be determined.
- *Cell ID* : <cell ID1>, <cell ID2>, <cell ID3>, ..., <cell IDn> is a set of cells identifiers. We suppose that each cell is identified by a single identifier.

We have $P_j(t_{r+1})$: the probability of the mobile terminal's location in the cell C_j at the time t_{r+1}.

We can calculate this probability by the following formula: $P_j(t_{r+1}) = \sum_{i=1}^{n} P_i(t_r) * P_{ij}$.

We define θ ($0 \leq \theta \leq 1$), which is a fixed or variable threshold. It is used to select the cells according to their probabilities. The MSpec is defined as follows:
$MSpec(t_r) = \{C_j / P_j(t_{r+1}) \geq \theta\}$.

Before the **m** associations, the system do not calculate the MSpec because the user is new and the system has not the necessary information to calculate the MSpec; it has no information concerning the M Matrix and the V Vector (observation phase).

The figure 8 represents the determination of the MSpec.

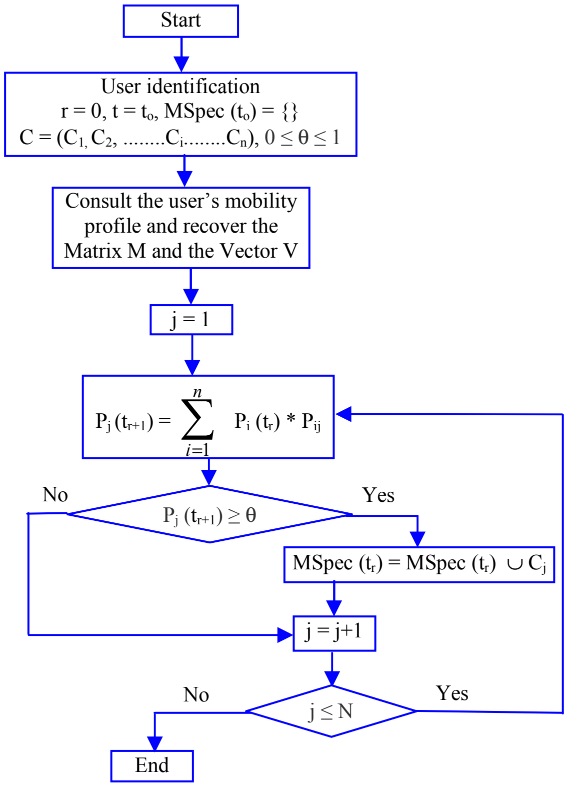


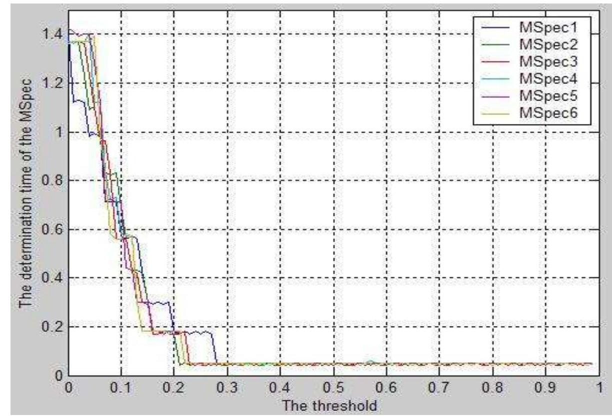
Figure 8. The MSpec determination

We use the MATLAB Mathematical Software to evaluate the impact of θ on the determination time of the MSpec and the impact of θ on the MSpec size. We suppose that the user's neighbourhood contains 10 cells and the value of m is 40 (the number of associations with the system).

During theses 40 associations, we will follow the different locations of the mobile terminal in order to determine the Matrix M and the Vector V.

After 40 associations with the system, the Matrix M and the Vector V are fixed.

The system calculates the vector $V_1 = V * M$ in order to determine the MSpec1 for the 1st handover. For the second handover, the system calculates the vector $V_2 = V_1 * M$ in order to determine the MSpec2 and so on.



This time doesn't exceed 1.5 millisecond.

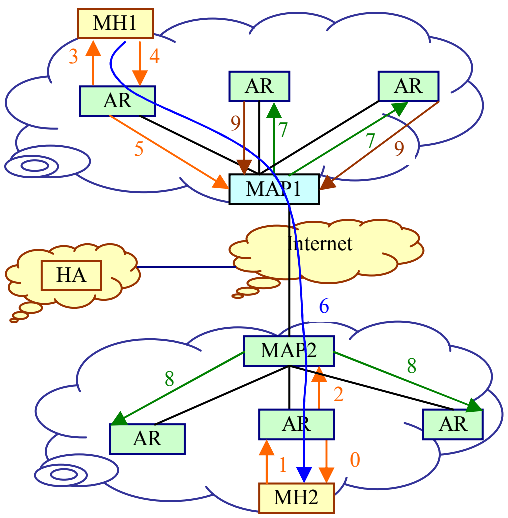
If $\theta \geq 0.28$, the MSpec is empty, we remark that $\theta = 0.1$ is a good value for the simulation.

6. Example

In the following, we present a scenario of communication between two mobile terminals where the MH1 is the entity, which generates the flow and the mode is *Sender Initiated Reservation.*

In this scenario, the two entities, which communicate are mobile and the MH1 generates the flow with the mode *Sender Initiated Reservation.* Therefore, the MH1 represents the NI (NSIS Initiator: The signaling entity, which makes the resource request, usually as a result of user application request), the MH2 represents the NR (NSIS Responder: The signaling entity that acts as the endpoint for the signaling and can optionally interact with applications as well) and the ARs as well as the MAPs represent the NF (NSIS Forwarder: The signaling entity between an NI end NR, which propagates NSIS signaling through the network).

We note MSpec1 and MSpec2 respectively, the set of future localisations of MH1 and MH2 during the communication.



MH: Mobile Host, AR: Access Router, MAP: Mobility Anchor Point,
HA: Home Agent, 0: Router Advertisement message (MIPv6),
1: Registration Request message (MIPv6),
2: NOTIFY message (QoS NSLP), 3: Router Advertisement message,
4: Registration Request message, 5: NOTIFY message,
6: RESERVE message (QoS NSLP), 7: NOTIFY message,
8: RESERVE message, 9: RESERVE message.

Figure 9. Advance resource reservation procedure

SLN NSLP is used to negotiate the level of service between MAP1 and MAP2 according to the user's QoS profile. This QoS profile contains the user's preferences in term of QoS for each application used. Then, MQoS NSLP is used to advance resources reservation.

Concerning the advance resources reservation using QoS NSLP, the procedure is as follows (the registration can start with the MH1 or the MH2, the following scenario considers that the MH2 is the first mobile, which makes the registration. The mode used is the *Sender Initiated Reservation* mode.):

0: The AR informs the MH2 with the message *Router Advertisement* of the availability of resources. For that, we propose to add a bit Q in this message. If Q = 0 then the AR does not have resources and in this case the MH2 can be connected in BE.

1: During the registration, the MH2 asks its AR for a certain QoS. In this case, we propose to add the MSpec2 object to the *Registration Request* message. (Here, we are interested only in the interactions between MIPv6 and the QoS NSLP messages, other MIPv6 messages are necessary in order to continue the registration).

2: After the registration with the MH2, the AR sends the QoS request to the MAP2. For that, we use the NOTIFY message with the MSpec2 object included in it. After the reception of the NOTIFY message, the MAP2 analyses the MSpec2 object.

3: The AR informs the MH1 with the *Router Advertisement* message of the availability of resources using the bit Q. If Q = 0 then the AR has not resources and in this case the MH2 can only be connected in BE.

4: during the registration, the MH1 asks its AR for a certain QoS. The MSpec1 object is added to the *registration request* message.

5: After the registration with the MH1, the AR sends the QoS request to the MAP1, for that we use NOTIFY message and the MSpec1 object. After the reception of the NOTIFY message, the MAP1 analyses the MSpec1 object.

6: To reserve the resources between the MH1 and the MH2, the MH1 (NI) sends the RESERVE message, which must contain the QSpec object. This message is transported by GIMPS until the MAP1, sent to the MAP2, to the AR and finally to the MH2 (NR).

7: After the reception of the RESERVE message, the MAP1 sends the NOTIFY message to all the ARs, which are in the MSpec1 in order to receive the RESERVE message.

8: the RESERVE message is forwarded after its reception by the MAP2, in all the ARs, which are in the MSpec2.

9: the ARs, which are in the MSpec1 sends the RESERVE message the MAP1.

The Handover procedure

The stages of the handover procedure are the following:

a. Registration of MH2 with its new AR (MIPv6 protocol).

b. Establishment of the new path and update of the resources reservation:

- b1. The new AR sends the RESERVE message to the MH2, (message 1 on the figure 10).
- b2. The MH2 sends the RESPONSE message with the new MSpec2, (message 2 on the figure 10).
- b3. After the reception of the RESPONSE message, the new AR sends the NOTIFY message to the MAP2 with the new MSpec2 (message 3 on the figure 10).
- b4. The MAP2 analyses the new MSpec2, and sends the corresponding RESERVE message (message 4 on the figure 10):

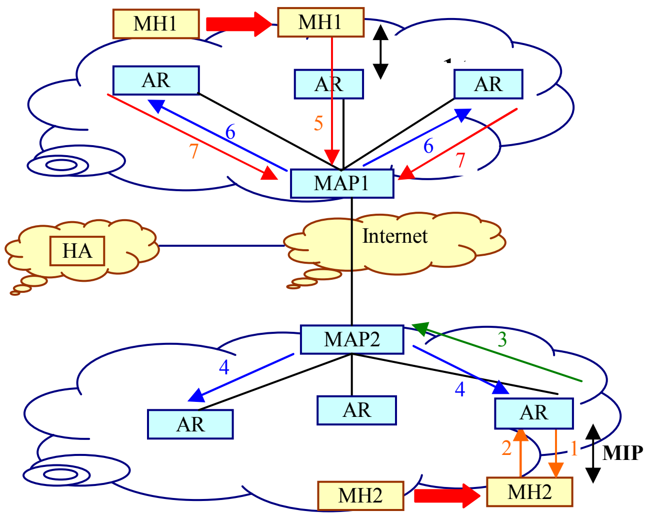
c. Registration of MH1 with its new AR (MIPv6 protocol).

d. Establishment of the new path and update of the resources reservation:

d1. The MH1 sends the RESERVE message to the new AR with the new MSpec1, it will be forwarded to the MAP1 (message 5 on the figure 10).

d2. The MAP1 includes the old and the new MSpec1 in a NOTIFY message. Then, it sends this message to all the ARs whose identification is in the new and the old MSpec1 (message 6 on the figure 10).

Each AR analyses the two MSpec1 objects and send the corresponding RESERVE message (message 7 on the figure 10):



1: RESERVE message (QoS NSLP),
2: RESPONSE message (QoS NSLP),
3: NOTIFY message (QoS NSLP),
4: RESERVE message, 5: RESERVE message,
6: NOTIFY message, 7: RESERVE message.

Figure 10. Handover procedure

7. Conclusion

In this paper, we propose a signaling environment for service level negotiation and advance resource reservation in mobile IP networks. This environment is built in conformance with the generic signaling environment, which is standardized by the NSIS IETF working group. Thus, SLN NSLP can be considered like an NSLP for the service level negotiation and so it uses the NTLP layer for the delivery of its messages and MQoS NSLP to reserve the resources corresponding to the negotiated SLS. We have presented a mobility and QoS profile used for service level negotiation and advance resource reservation in a mobile environment. This reservation is made according to the MSpec object which determines the future locations of the mobile terminal**.** Our objective through this approach is to minimize the degradation of services during the handover. We have presented a scenario when the two terminals are mobile. Future works concern the implementation of this environment in the context of the National French project called SWAN [15].

8. Bibliography

[1] S. Van den Bosch, G. Karagiannis, A. McDonald, "NSLP for Quality-of-Service signaling", <draft-ietf-nsis-qos-nslp- 10.txt>, March, 2006.

[2] Anup Kumar Talukdar, B. R. Badrinath, Arup Acharya "MRSVP: A resource reservation protocol for an integrated services network with mobile hosts", ACM Journal of Wireless Networks, vol. 7, 2001.

[3] Anup Kumar Talukdar , B. R. Badrinath , Arup Acharya, "Integrated services packet networks with mobile hosts": architecture and performance, Wireless Networks, v.5 n.2, p.111-124, March 1999.

[4] Anup Kumar Talukdar, B. R. Badrinath , Arup Acharya, "On Accommodating Mobile Hosts in an Integrated Services Packet Network", Proceedings of the INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution, p.1046, April 09-11, 1997.

[5] Min-Sun Kim, Young-Joo Suh, Young-Jae Kim, Young Choi. "A Resource Reservation Protocol in Wireless Mobile Networks". ICPP Workshops, Valencia, Spain September 03 - 07, 2001.

[6] Thi Mai Trang N., Boukhatem N.,Ghamri Y., Pujolle G. «Service Level Negotiation and COPS-SLS Protocol», *IEEE Communication Magazine,* May 2002, pp. 158-165

[7] Chen J., McAuley A., Sarangan V., "Dynamic Service Negotiation Protocol", IETF, draft-dsnp-01, December 2002.

[8] Mescal D1.2 "Initial specification of protocols and algorithms for inter-domain SLS management and traffic engineering for QoS-based IP service delivery and their test requirements", January 2004. Available: http://www.mescal.org

[9] Olivier Dugeon, «From SLA to SLS up to QoS control : The CADENUS Framework», 2002. Available: http://www.cadenus.org

[10] Goderis D., Griffin D., «Attributes of a service level spécification template», IETF, draft-tequila-sls-03, October 2003.

[11] Benmammar B., Thi Mai Trang N., Pujolle G., Yilmaz V., « Définition d'un SLA / SLS », IP-SIG Document, December 2003.

[12] Brunner M., « Requirements for Signalling Protocols », IETF, RFC 3726, Avril 2004.

[13] H. Schulzrinne, R. Hancock, "GIST: General Internet Signaling Transport", draft-ietf-nsis-ntlp-09, February, 2006

[14] Ash J., Bader A., Kappler C. « QoS NSLP QSpec Template », IETF, draft-nsis-qspec-09, March 2006.

[15] SWAN, "Self aware management", RNRT French project, Available: http://www.telecom.gouv.fr/rnrt/rnrt/projets