

A User Interface for Resource Management in a Mobile Environment

Badr Benmammour, Zeina Jrad, Francine Krief

► **To cite this version:**

Badr Benmammour, Zeina Jrad, Francine Krief. A User Interface for Resource Management in a Mobile Environment. The IFIP/IEEE International Conference on New Technologies, Mobility and Security (NTMS 2007), May 2007, Paris, France. 2007. <hal-00657681>

HAL Id: hal-00657681

<https://hal.inria.fr/hal-00657681>

Submitted on 9 Jan 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A User Interface for Resource Management in a Mobile Environment

Badr Benmammour¹, Zeina JRAD² and Francine Krief³

¹GET / Télécom Paris CNRS LTCI
46 rue Barrault - 75634 Paris Cedex 13
benmamma@enst.fr

WWW home page: <http://www.ltcn.enst.fr/>

²INRIA Rocquencourt, Domaine de Voluceau
B.P. 105, 78153, Le Chesnay, France
zeina.jrad@inria.fr

WWW home page: <http://www.inria.fr/roquencourt/>

³LaBRI Laboratory, Bordeaux1 University, 33400 Talence, France
francine.krief@labri.fr

WWW home page: <http://w5.labri.fr/>

Abstract: This paper describes a user interface for QoS management in mobile IP networks. The paper context is built in conformance with the generic signaling environment, which is standardized by the NSIS IETF working group. In this work, we investigate the use of some techniques of the AI (Artificial Intelligence) domain to implement a user interface called NIA (Negotiation Individual Assistant) in order to determine the QoS profile and negotiate the QoS parameters in the new domain after the handover. Therefore, we use the connectionist learning in the management of the negotiation profiles and the agent technology to help the user to choose the best service provider, dynamically negotiate the QoS on the user's behalf, follow the user's behaviour to be able to anticipate the negotiation and manage the re-negotiation. The resource management, presented in this work, provides to mobile terminals the required QoS based on user's mobility and QoS profile. This QoS profile is determined by the NIA.

1 Introduction

The development of real time applications as well as multimedia applications has witnessed an exponential increase. The real time constraints of these applications present a big challenge for their integration. That's why we need services adapted to specific application needs with a guaranteed Quality of Service (QoS) [1].

However, the implementation of QoS mechanisms is a very heavy task. It is difficult to manually configure all the network devices because of the abundance of QoS information and because of the dynamic nature of QoS configurations. The operator must control the attribution of network resources according to applications and users characteristics. Using management tools adapted to QoS quickly proves essential. In order to simplify the router's configuration by permitting its automation, the IETF proposed a general framework called policy-based networking [2] for the control and management of these IP networks. The Internet New Generation is based on the policy-based networking management.

Most applications cannot dynamically express their QoS requirements to obtain the adapted level of service. For each application, the customer and the provider have to agree on rules of assignment of service levels. They sign a contract called SLA (Service Level Agreement) which is then translated into high-level policies. These policies are not directly executable by the network devices. They must be translated into intermediate and then into low level policies which are understandable by network devices. The SLS (Service Level Specification) is the technical version of the SLA [3] and the QoS parameters (also called performance metrics) are a part of the SLS parameters. The negotiation of these parameters is a difficult process because the user needs to identify himself their values according to his context. The user interface, called NIA (Negotiation Individual Assistant), presented in this paper is supposed to determine using artificial intelligent techniques, a context-aware profile for the QoS negotiation.

On the other hand, 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 [4] signaling application.

The objective of our work is to propose a signaling environment for QoS negotiation and advance resource reservation in mobile IP networks in conformance with the generic signaling environment standardized by the NSIS IETF working group. The QoS negotiation and resource reservation is based on the profile determined by the NIA (Negotiation Individual Assistant).

This paper is organized as follows. The first section presents a synthesis of the research relating to resources reservation in an IP mobile environment. The second section defines the NSIS environment in which we specify the advance resource reservation protocol. The third section presents our user's mobility and QoS profile. Then, we present the user interface for profile determination. The last section is a description of an utilization case of the signaling environment.

2 QoS in Mobile IP Networks

Recent research takes an interest in advance resource reservation to provide the necessary QoS to the mobile terminals.

The authors in [5] proposed a new protocol of resource reservation in a mobile environment called MRSVP (Mobile RSVP). 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 new 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.

Min-Sun Kim et al. [6] 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 by the best effort flows.
- *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.

Ferrari et al. [7] described a distributed mechanism in order to make reservations in advance for the real time connections. In this mechanism, the reservation demand is classified according to two types: *immediate* and *in advance*.

- An Immediate reservation is activated at the moment of the demand; its length is not specified.
- A reservation in advance is associated with two parameters: *starting time* (the time of the reservation activation) and *duration* (the reservation period).

Another way to obtain a better use of resources is to determine the future locations of the mobile terminal.

3 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.

3.1 GIST

GIST (General Internet Signaling Transport) is the protocol that NSIS has adopted as a standard for the NTLP layer [8]. 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.

3.2 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 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.

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.

3.3 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 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.

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

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

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 and is determined by the NIA (Negotiation Individual Assistant).

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: $P_{ij} = t [i, j] / g (i)$.

- $V = [P_i (t_0)] [N]$: This Vector contains the $P_i (t_0)$.

We note:

$P_i (t_0)$: this probability defines the location of the mobile terminal in the cell C_i at the time t_0 .

$k (i)$: the number of association with the cell i during the m associations at the time t_0 .

We have: $\sum_{i=1}^n k (i) = m$ and $P_i (t_0) = k (i) / m$.

- 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:

$$MSpec(t_r) = \{ C_j / P_j(t_{r+1}) \geq \theta \}.$$

5 User Interface for QoS negotiation

The user interface proposed in this paper is placed on the user terminal and is called NIA (Negotiation Individual Assistant). The NIA negotiates the QoS between the user and the service provider, from one side, and between the user and the network, from the other side (figure 1). The main purpose of the interface is the representation of the user in requesting and negotiating the desired QoS in a dynamic environment.

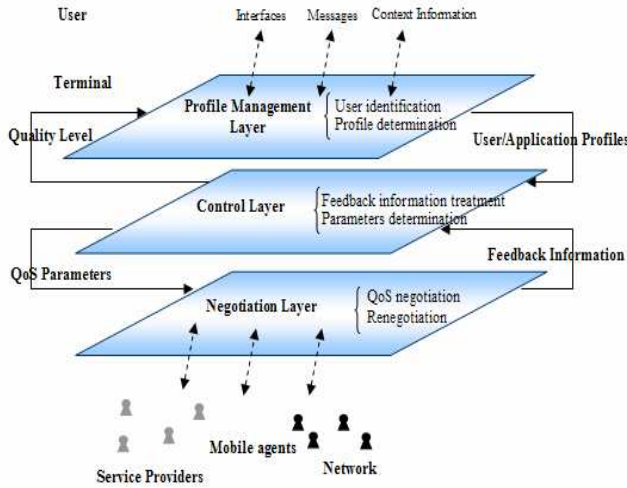


Fig. 1. The layers of the proposed framework

As shown in figure 1, the proposed interface contains different layers. The first one is the profile management layer. This is the layer of direct contact with the user. Once connected, the principal task of the modules of this layer is to react autonomously in order to follow the user’s work. User and terminal contexts are saved along with the used applications and their requirements in the knowledge base of the system. These data are modified systematically according to any change in the user choices and actions. The reasoning modules will also use them in order to deduce a general profile that represents the user.

The second layer is for the control. Once the user’s needs and preferences are identified, the next step consists of verifying that these preferences are converted into the appropriate SLS values.

The third layer is the Negotiation layer. This layer manages the service publication, subscription, selection and negotiation.

In this paper, we will focus on the first layer, further details concerning the second and the third layers can be found in [17, 18].

5.1 Profile Management Layer

This layer is introduced to identify the user and to analyze his work. User preferences and application requirements are saved in the knowledge base of the system. These data are modified systematically according to any change in the user choices and actions. Applications can be classified in many categories according to their needs (delay, jitter...) and to the type of the supported information (data, voice, image...). The profile of the application will then be determined according to these categories and to the requirements of the user.

As an input to this layer, we can identify two types of data:

- Information collected through a communication with the user via graphical interfaces or messages.
- Information collected through an observation of the user's behavior.

Once the information is analyzed, the result is a user profile that represents the user's preferences in terms of quality and an application profile that describes the needs of each application.

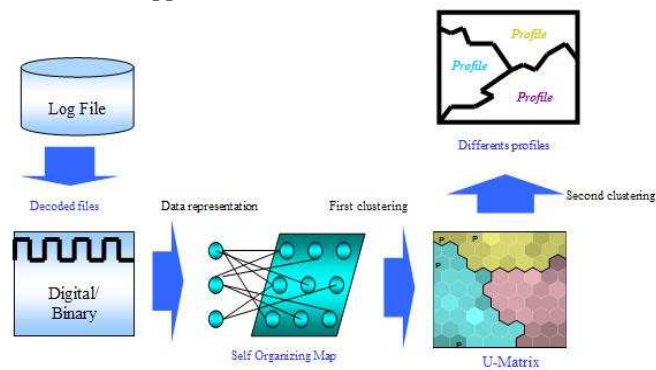


Fig. 2. Profile management procedure

Our approach, represented in figure 2, consists in recovering, first of all, data that represent traces of use (i.e. log files [9]). These data will be cleaned and re-coded in a numerical or binary format to be easily treated. We build then a Self Organizing Map from the re-coded file in order to extract profiles [10]. Finally, we carry out a classification to better see the clusters structure of the map, followed by a segmentation of the SOM in order to separate the different profiles.

The steps of our approach represented in figure 2 are detailed in the following subsections.

5.1.1 The unsupervised connectionist learning

The unsupervised numerical learning, or automatic classification, consists in determining a partition of an instances space from a given set of observations, called training set. It aims to identify potential trend of data to be gathered into classes. This kind of learning approach, called clustering, seeks for regularities from a sample set without being driven by the use of the discovered knowledge. Euclidian distance is usually used by clustering algorithms to measure similarities between observations.

Self-Organizing Maps (SOM) implement a particular form of competitive artificial neural networks [10]; when an observation is recognized, activation of an output cell – competition layer – leads to inhibit activation of other neurons and reinforces itself. It is said that it follows the so called “Winner Takes All” rule. Actually, neurons are specialized in the recognition of one kind of observations. The learning is unsupervised because neither the classes nor their numbers are fixed a priori.

The training set is used to organize these maps under topological constraints of the input space. Thus, a mapping between the input space and the network space is constructed; closed observations in the input space would activate closed units of the SOM.

An optimal spatial organization is determined by information received from the neural networks. When the dimension of the input space is lower than three, both of the position of weights vectors and the direct neighborhood relations between cells can be visually represented. Thus, a visual inspection of the map provides qualitative information on its architecture.

The connectionist learning is often presented as a minimization of a risk function [11]. In our case, it will be carried out by the minimization of the distance between the input samples and the map prototypes (referents), weighted by a neighborhood function h_{ij} . To do that, we use a gradient algorithm. The criterion to be minimized is defined by:

$$E_{SOM} = \frac{1}{N} \sum_{k=1}^N \sum_{j=1}^M h_{jNN(x^{(k)})} \|w_{.j} - x^{(k)}\|^2$$

Where N represents the number of learning samples, M the number of neurons in the map, $NN(x^k)$ is the neuron having the closest referent to the input form x^k , and h the neighborhood function. The neighborhood function h can be defined as:

$$h_{rs} = \frac{1}{\lambda(t)} \exp\left(-\frac{d_1^2(r,s)}{\lambda^2(t)}\right)$$

$\lambda(t)$ is the temperature function modeling the neighborhood extent, defined as:

$$\lambda(t) = \lambda_i \left(\frac{\lambda_f}{\lambda_i} \right)^{\frac{t}{t_{\max}}}$$

λ_i and λ_f are respectively initial and the final temperature (for example $\lambda_i=2, \lambda_f=0.5$). t_{\max} is the maximum number allotted to the time (number of iterations for the x learning sample). $d_1(r,s)$ is the Manhattan distance defined between two neurons r and s on the map grid, with the coordinates (k,m) and (i,j) respectively:

The learning algorithm of this model proceeds essentially in three phases:

- Initialization phase where random values are assigned to the connections weights (referents or prototypes) of each neuron of the map grid.
- Competition phase during which, for any input form $x^{(k)}$, a neuron

$NN(x^{(k)})$, with neighborhood, $V_{NN(x^{(k)})}$ is selected like a

winner. This neuron has the nearest weight vector by using Euclidean distance:

$$d_1(r, s) = |i - k| + |j - m|$$

$$NN(x^{(k)}) = \underset{1 \leq i \leq M}{\operatorname{argmin}} \|w_i - x^{(k)}\|^2$$

- Adaptation phase where the weights of all the neurons are updated according to the following adaptation rules:

If $w_j \in V_{NN(x^{(k)})}$ then adjust the weights using:

$$w_j(t+1) = w_j(t) - \varepsilon(t) h_{jNN(x^{(k)})} (w_j(t) - x^{(k)})$$

Else

$$w_j(t+1) = w_j(t)$$

Repeat this adjustment until the SOM stabilization.

5.1.2 SOM map segmentation

We segment the SOM using the K-means method (Figure 3). It is another clustering method that consists in arbitrarily choosing a partition; the samples are then treated one by one. If one of them becomes closer to the center of another class, it is moved into this new class. We calculate the centers of new classes and we reallocate the samples to the partitions. We repeat this procedure until having a stable partition.

The criterion to be minimized in this case is defined by:

$$E_{K-means} = \frac{1}{C} \sum_{k=1}^C \sum_{x \in Q_k} \|x - c_k\|^2$$

Where C represents the number of clusters, Q_k is the cluster k , C_k is the center of the cluster Q_k or the referent.

The basic algorithm requires fixing K , the number of wished clusters. However, there is an algorithm to calculate the best value for K assuring an optimal clustering. It is based principally on the minimization of Davies-Bouldin index [12], defined as follows:

$$I_{DB} = \frac{1}{C} \sum_{k=1}^C \max_{l \neq k} \left\{ \frac{S_c(Q_k) + S_c(Q_l)}{d_{ce}(Q_k, Q_l)} \right\}$$

With,

$$S_c(Q_k) = \frac{\sum_i \|x_i - c_k\|^2}{|Q_k|}$$

$$d_{cl}(Q_k, Q_l) = \|c_k - c_l\|^2$$

C is the number of clusters, S_c is the intra-cluster dispersion, and d_{cl} is the distance (centroid linkage) between the clusters centers k and l . This clustering procedure aims to find internally compact spherical clusters which are widely separated.

There are several methods to segment the SOMs [13]. Usually, they are based on the visual observations and the manual assignment of the map cells to the clusters. Several methods use the K-means algorithm with given ranges for K value. Our work is based on the approach of Davies-Bouldin index minimization.

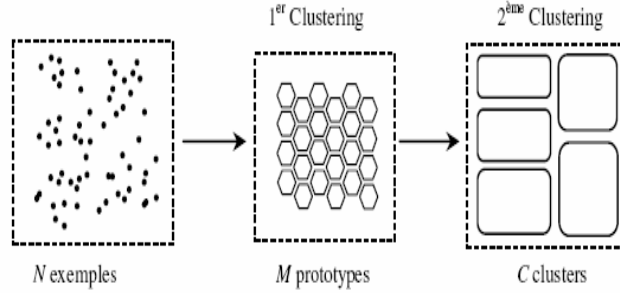


Fig. 3. Two successive clusterings: SOM followed by K-means

We note that the K-means approach can be directly applied to the data instead of SOMs approach. In our work, we applied it to the SOMs results. The idea is to use SOMs as a preliminary phase in order to set a sort of data pretreatment (dimension reduction, regrouping, visualization...). This pretreatment has the advantage to reduce the clusters calculation complexity and also ensures a better visualization of the automatic classification results.

5.1.3 Simulations results

We applied the two algorithms described above on our data (log files describing different traces of use) in order to determine the negotiation profiles. In the simulations, we used the *SomToolbox* proposed by the researchers of the HUT (Helsinki University of Technology) of the T. Kohonen team [14]. The results obtained are very promising (Figure 4).

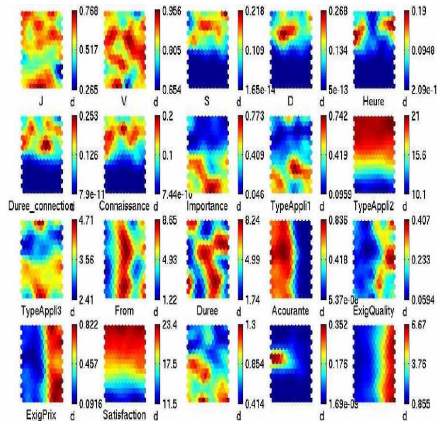


Fig. 4.a. SOM clustering

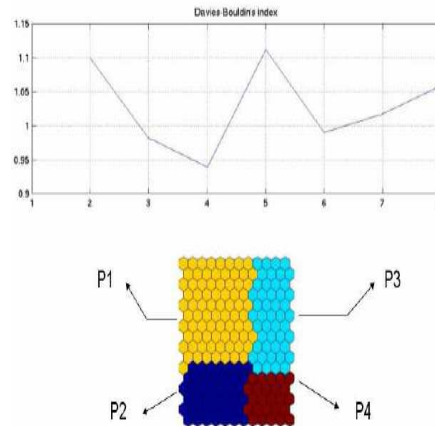


Fig. 4.b. Resulting profiles

Figure 4.a is a representation of a SOM map seen as “*Component Planes*” that allows the visualization of the partition of the different variable values. The highest values of the variables are in red and the lowest values are in blue.

Figure 4.b represents the neurons segmentation of SOM map (second clustering). The curve shows that the minimal value of the index of Davies-Bouldin corresponds to an optimal clustering resulting in four profiles. The various colors represent the different clusters or the identified negotiation profiles.

6 Conclusion

In this paper, we propose a user interface for QoS management in mobile environment. We have presented a mobility and QoS profile used for QoS negotiation and advance resource reservation in a mobile IP network. 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. The Negotiation Individual Assistant (NIA) presented in this paper constitutes an interface between the user and the network in the context of the Internet of new generation. This interface integrates two interesting techniques of the Intelligence Artificial domain: The connectionist learning and the agent technology. Our NIA has been implemented and tested in two French national research projects: ARCADE [15] and IPSIG [16].

References

1. QoSforum, "QoS protocols & architectures", White Paper, July 1999.
2. Yavatkar R., D. Pendarakis, R. Guerin, "A Framework for Policy Based Admission Control", RFC 2753, January 2000.

3. D. Goderis and al., Service Level Specification Semantics and Parameters, draft-tequila-sls-02.txt, Internet draft, January, 2002.
4. S. Van den Bosch, G. Karagiannis, A. McDonald, NSLP for Quality-of-Service signaling, <draft-ietf-nsis-qos-nslp- 10.txt>, March, 2006.
5. A. K. Talukdar, B. R. Badrinath, A. Acharya, MRSVP: A resource reservation protocol for an integrated services network with mobile hosts, ACM Journal of Wireless Networks, vol. 7, 2001.
6. K. Min-Sun, S. Young-Joo, K. Young-Jae, C. Young, A Resource Reservation Protocol in Wireless Mobile Networks. ICPP Workshops, Valencia, Spain September 03 - 07, 2001.
7. D. Ferrari, A. Gupta, G. Ventre, Distributed Advance Reservation of Real-Time Connections. Proceedings of the 5th International Workshop on Network and Operating System Support for Digital Audio and Video. P.16-27. April 19-21, 1995.
8. H. Schulzrinne, R. Hancock, GIST: General Internet Signaling Transport, draft-ietf-nsis-ntlp-09, February, 2006.
9. K. Benabdeslem, Approches Connexionnistes pour la visualisation et la classification des données issues d'usages de l'Internet, thesis in computer science, LIPN, University of Paris13, France, December, 2003.
10. T. Kohonen and S. Kaski and H. Lappalainen, Self-Organized Formation of Various Invariant-Feature Filters in the Adaptive-Subspace SOM, pages 1321-1344, Neural Computation, 1997.
11. Y. Bennani, Réseaux de neurones artificiels, Chapter in Encyclopedie d'Informatique et Science de l'information, Edition Vuibert, 2005.
12. Davies, D., L., Bouldin, D., W., A Cluster Separation Measure, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1(2): pp. 224-227. (1979).
13. Juha, A. , Esa, A., Clustering of the Self-Organizing Map, IEEE Tractions On Neural Networks volume 11, n° 3, (2000).
14. E. Alhoniemi and J. Himberg and J. Parhankangas and J. Vesanto, SOM Toolbox, 2000, Copyright (C),<http://www.cis.hut.fi/projects/somtoolbox/>.
15. Projet Arcade, Architecture de Control Adaptative des Environments IP, Web Site: <http://www-rp.lip6.fr/arcade/>, 2002.
16. Projet IpSig, Signalisation générique du monde IP, http://www.telecom.gouv.fr/rnrt/rnrt/projets/res_02_85.htm, 2004.
17. Zeina JRAD, Thesis report, Apports des techniques de l'Intelligence Artificielle dans la négociation dynamique de la qualité de service : Proposition d'un assistant à l'utilisateur dans les réseaux IP de nouvelle génération, LIPN, University of Paris 13, France, May 2006.
18. Z. Jrad., B. Benmammar, J. Correa, F. Krief , N. Mbarek, A user assistant for QoS negotiation in a dynamic environment using agent technology. Second IFIP International Conference on Wireless and Optical Communications Networks WOCN 2005. Dubai, United Arab Emirates UAE, March 2005.