Integrating Networks Measurements and Speech Quality Subjective Scores for Control Purposes

Samir Mohamed[a], Francisco Cervantes-Pérez[1b], and Hossam Afifi[c]

[a] INRIA/IRISA, Campus de Beaulieu 35042, Rennes cedex, France.
Samir.Mohamed@irisa.fr
[b] Instituto Tecnológico Autónomo de México (ITAM), Mexico.
Cervante@itam.mx
[c] RST Dept., INT Evry, rue Charles Fourier, 91011 Evry, France.
Hossam.Afifi@int-evry.fr

Abstract — **Traditionally, QoS has been addressed by using network measurements (e.g., loss rates and delays), and little attention has been paid to the quality perceived by end-users of the applications running over the network. Here, we address the issue of integrating speech quality subjective scores and network parameters measurements, for designing control algorithms that would yield the best QoS that could be delivered under a given communications network situation. First, we build a neural network based automaton to *measure* speech quality in real time, at the *style* of a group of human subjects when participating in an MOS test. We consider the effects of changes in network parameters (e.g., packetization interval, packet loss rate and their pattern distribution) and encoding on speech signals transmitted over the network. Our database includes transmitted speech signals in different languages. Then, we outline a control mechanism which, based on the application performance within a session (i.e., MOS speech quality scores generated by the neural networks), dynamically adjusts parameters (codec and packetization interval). Finally, we analyze preliminary results to show two main benefits: first, a better use of bandwidth, and second, delivery of the best possible speech quality given the network current situation.**

Index Terms — **Voice over IP, Packet Switched Networks, Speech Quality Assessment, Neural Networks and End-to-End Control Mechanisms.**

I. Introduction

Recent studies on real-time multimedia applications [14][20][23][30] involving the transmission of voice over packet switched networks have emphasized the difficulties of having a best-effort Internet service model. It introduces variable delays and loss distribution patterns that greatly decrease audio quality. This best-effort delivery policy will not change for a long time; thus, in order to get acceptable Quality of Service (QoS) levels, it has become extremely important to develop control mechanisms that eliminate, or at least minimize, the effects of network parameters on the quality of audio signals as perceived by users at destination points [2]. In fact, the quality perceived by end-users may define the scope of applicability [20], or, furthermore, the final acceptance of real-time multimedia services [28]. For example, the QoS delay constraint could be mitigated when using Internet telephony on a local or metropolitan basis.

Several approaches have been developed to address this issue, including:

a) forward error correction (FEC) algorithms have been developed to minimize the effect of packet loss, by sending additional information to aid in packet recovery [2][23][25];
b) control mechanisms working at destination points to minimize the effect of delay variations between successive packets [3][8];
c) adapting scalable bit-rate codec and prioritized transmission algorithms, at the network layer, to get a smooth degradation of quality during network congestion periods [1];
d) equation-based control algorithms to reduce congestion for unicast applications [12].

Traditionally, control algorithms to improve QoS have been designed based on measurements of a wide range of parameters, the most commonly used being: network parameters (packet loss rate, packetization interval, arrival jitter, loss distribution, and end-to-end delay); parameters related to the ability of codecs to encode or compress original speech signals without losing significant information (codec algorithm, sampling rate, and number of bits per sample); and other parameters like the echo (which may occur due to long end-to-end delay), the crosstalk effect (when two or more persons start talking at the same time), and the number of participating sources in the multiparty conferences [18][21]. Here, we focus only on unidirectional communication processes.

Even though the main concern for QoS is to maximize the quality of the data delivered at the destination point, most of the current discussion is concentrated on finding a way to keep within certain limits some of the network parameters (e.g., packet loss rate (LR) and delay variation), and little attention has been paid to the quality perceived by the end-

[1] F. Cervantes-Pérez is on sabbatical leave at the *Ecole Nationale Supérieure des Télécommunications de Bretagne* (ENSTB), Campus Rennes, France.

users on the application running over the network. Although the average perceived quality at the destination points has been used to analyze how changes in network parameters, due to the action of control algorithms, improve speech quality [2], it has not been used to actually design the control algorithms.

Other authors have shown the advantages of including an application's semantics in the design of that application protocol, like in Application Level Framing (ALF) [5], and in the Scalable Reliable Multicast (SRM) framework [11]. Thus, in a similar fashion, in order to obtain the best QoS that can be offered given a particular network situation, we propose to develop a control mechanism that takes into account the assessment of speech quality itself, as would be perceived by users at the destination point during the execution of a specific application. However, before this can be done, the issue of speech quality assessment in real time needs to be tackled. The main difficulty is that, although there are two types of methods to measure speech quality, both involve time-consuming procedures, which makes it impossible to implement them in a real-time application. The first method is objective quality assessment, in which speech quality assessment is carried out based on mathematical analyses that compare original and distorted samples [9][10][13][24]. The second method is subjective quality assessment, where a group of people, acting as users at the destination point, give a measure of an application performance by rating speech signals distorted by different situations of the network over which they are transmitted [17][19][20].

Based on the fact that, so far, the best way to evaluate speech quality is subjective listening tests [29], in a preliminary study [20] we showed how neural networks could be used to model humans' subjective assessment of speech quality during a Mean Opinion Score (MOS) listening test. In this paper, our aim is to continue the development of a neural networks based automaton to measure speech quality in real-time, by including a more robust database to train the neural networks. The data was gathered by conducting MOS listening tests with groups of people assessing speech quality in different languages (Spanish, Arabic and French).

An additional motivation of our work is to lay the basis for designing control mechanisms that integrate measurements of network parameters with the *subjective* speech quality scores generated by the neural networks. These mechanisms should work in real time, and they should integrate the neural networks quality score with other network parameters (e.g., encoding, packetization interval, packet loss rate and their pattern distribution) and other control mechanisms (e.g., FEC, congestion control), in order to guarantee the best possible speech quality given the communications network current situation.

Our paper is organized as follows: in Section II, we describe the testbed used to measure and select the important parameters and their ranges in order to train the neural network. Section III addresses the collection of the Mean Opinion Score database and the result analysis. Section IV exhibits the training phase of the neural network and shows the ability of the neural network to reproduce the MOS scoring. Finally, Section V proposes a new real-time control mechanism where TCP-Friendly congestion control is combined with the neural network to obtain better speech quality and to save additional bandwidth.

II. Measuring Network Parameters in a Testbed

This section presents the testbed that we used to identify the most relevant parameters and their ranges for real-time transmission of speech. In addition to the parameters mentioned in the introduction, we noticed that real-time audio in general and speech in particular are sensitive to other factors that, when combined with the network behavior, give a different resulting perceived quality. One of these parameters is the spoken language [19]. A series of tests with different languages showed in fact that they do not equally tolerate the losses. More refined analysis that was not however taken into account in our development showed that the speaker speech style is also an important parameter in the comprehension tests. For these reasons, all our MOS experiments were done for three languages.

A simple survey on IP telephony users today showed that they generally fall into the following categories. A first category concerns telephony nationwide. We did for that reason, three series of measurements between Rennes (ENST-B) and peers laying respectively in Rennes (Irisa) (2 Km distance), Brest (300 Km) and Sophia Antipolis (1300 Km). The second category is related to international phone calls and for that reason tests were accomplished between Rennes and Mexico City (Mexico). The number of hops, minimum, average, and maximum one-way delay in ms between Rennes (ENST-B) to the other sites are shown in Table 1.

The third category of communications in a LAN was not considered because it was difficult to find a real practical situation where a 10Kb/s IP flow does not easily find its way through a 10Mb/s or even a 100Mb/s LAN.

Each session consists of sending a 160-byte packet every 20 ms as real-time traffic carried by RTP protocol. The duration of the session is 10,000 packets. The receiver reports the total number of packets, the total loss rate, and

TABLE I
NUMBER OF HOPS AND ONE-WAY DELAY STATISTICS.

Site	Hops	Min. delay	Av. delay	Max. delay
Irisa	6	4.2	11	70
Brest	7	7.1	43.3	117
Sophia	12	24	35	60
Mexico	28	149	159	221

percentage rate of each *n*-consecutively-lost packets. The destination considers any packet that arrived after the playback threshold as lost. This is to avoid the jittering problem. In fact, there are several algorithms to choose the best value of the playback threshold in order to minimize the percentage loss rate and to avoid the jittering problem [8].

For each site, we repeated (50 times for each site) the tests in working-day hours and in different days. Then we selected the results that give the maximum and minimum percentage loss rate. By varying the playback time length, the percentage loss and the loss distribution change accordingly. Fig. 1 depicts the minimum and maximum percentage loss rate. As expected, the loss rates decrease when we increase the playback buffer length of the receiver and the number of hops. In Fig. 2 and Fig. 3 we plot the rates of consecutively lost (CL) packets against the buffer size. The first figure shows minimum values and the second shows maximum ones. The two figures show the frequency of n-consecutively lost packets where n varies from one to ten.

As it is clear from these two figures, in national sites, the three consecutive loss pattern is considered the limit. While in the international case, five consecutive losses is the limit.

The same approach was taken in [4] (with almost the same paths for tests), however the number of years separating the two experiments clearly show an improvement in delay and loss bounds.

The tests that we described were intended only to give a realistic figure on the average values taken by the parameters that we use in the neural networks training. Therefore, the databases that will be used to train the neural network are as close as possible to real network situations.

Based on the statistics collected from the above experiment, we have selected values for the network parameters as follows:

1. Loss rate: 0, 5, 10, 20 and 40 %. In fact, the loss rate depends on the bandwidth, network load, congestion and the choice of a strict playback time. For more details about packet loss dynamics, see [23].
2. Loss distribution: we have chosen the number of consecutively lost packets as the loss distribution, which varies from 1 up to 5 packets dropped at a time. Of course in some situations, there may be more than 5-consecutively-lost packets. However, for the sake of simplicity, we consider them as multiple of 5-consecutively-lost packets.
3. We have chosen the values of the packetization intervals as 20, 40, 60 and 80 ms. In fact the majority of the existing real-time applications use one or more of these values.
4. For the speech encoding algorithms, we have selected PCM (64kbps), G726 ADPCM (32kbps) and GSM-FR (13.2kbps). The corresponding packet size is 160, 80, and 33 bytes for a 20 ms packetization interval. From the literature [15], the corresponding subjective MOS quality rates are 4.4, 4.1 and 3.6 respectively. These values correspond to an absolute score that evaluates the codec with non-distorted samples.

Of course, one can use any mechanism of FEC [7] to reduce the amount of total loss but on the cost of extra delay, which in turn controls the loss rate if strict playback time mechanism is used. In networks employing some kind of QoS mechanisms like IP DiffServ or ATM, one can control the amount of losses to achieve a certain level of audio/speech quality [1].

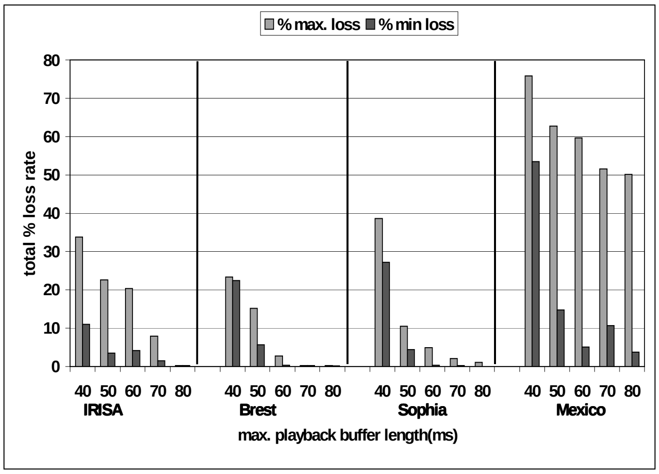

Fig. 1. Maximum and minimum percentage loss rates.

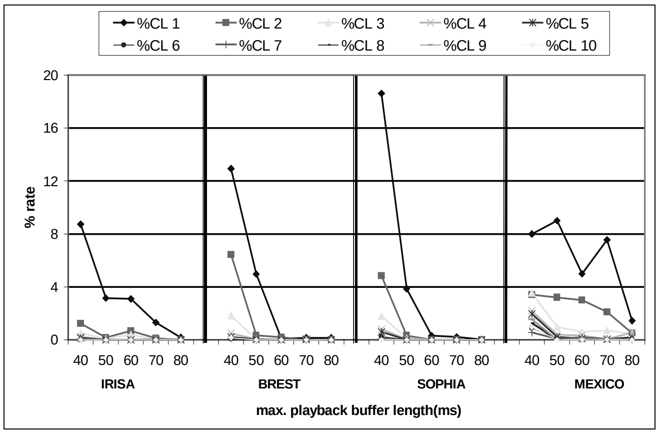

Fig. 2. Minimum rates for one-to-ten consecutively lost packets.

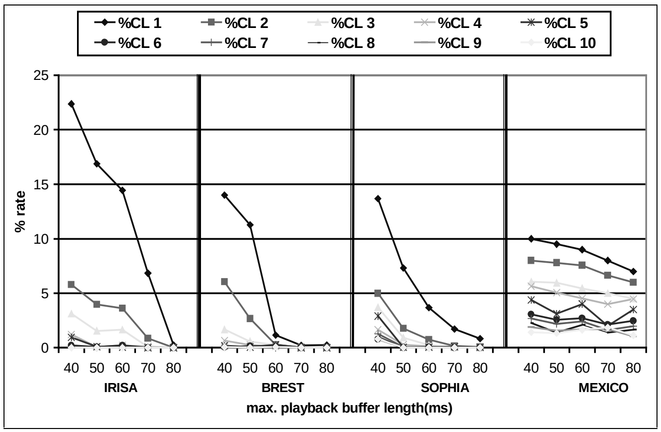

Fig. 3. Maximum rates for one-to-ten consecutively lost packets.

A. Other Effects

The delay will map to loss if a strict playback time mechanism is used. Moreover, when a dejittering buffer [8] is implemented, the effect of jitter will be masked and it will map to loss. In one-way sessions, there is no echo effect [7]. However, when two-way sessions are used, one should use a mechanism to control the echo effect, like echo suppression or echo cancellation. For a complete discussion for packet loss, delay jitter and the recovering mechanisms, see [14].

III. A Mean Opinion Score (MOS) Database for Different Languages

In building neural networks applications, the first step is data collection. Here, we collected a database conformed by a set of samples of distorted speech signals, according to network parameters variations as described in Section B, and their associated quality score given as the average opinion of a group of human subjects.

The quality scores were obtained through a series of MOS experiments, which were carried out following an ITU-recommended method [17]. It should be noted that some authors [28] have questioned the suitability of these methods to assess speech and video quality when transmitted over packet networks. However, in our approach, whenever a better method to evaluate speech quality becomes a standard, in order to update our system it will only be required to repeat the MOS experiments and to retrain the neural networks with the new database.

Distorted speech signals were generated by using an IP network tesbed, comprised by a sender, a router, and a receiver. The sender controls the packetization interval, the selection of the encoding algorithm and the sending rate. The router controls the loss rate, and the loss pattern distribution. Finally, the receiver stores the received signals, decodes them, substitutes lost packets by silence periods, and it calculates the loss rate and number of consecutively lost packets. Of course, one can substitute lost packets by comfort noise, waveform substitution, etc. For simplicity we chose silence insertion.

Each sample consists of speech material sent from the sender to the receiver according to a given combination of network parameters values obtained in Section B and in data from the literature. In addition, it has been established that coders performance shows language dependency [19]. Thus, language type is another variable included in our database. We conducted MOS tests in three different languages: Spanish, Arabic and French. During the tests, groups of human subjects (15 in the Spanish and Arabic tests, and 7 in the French test) were asked to indicate the quality of the speech they heard on a 5-point MOS scale.

Our results show that speech quality degrades in all languages as the packet loss rate increases, regardless of the packetization interval, language and codec (see Fig. 4 and 5)

In addition, under certain situations, speech quality language dependency was also observed. That is, given the same network parameter variations, the MOS scores differ according to the languages. For example, when encoding with PCM, using a packetization interval of 20 *msec.*, and having a packet loss rate of 5%: a) in Spanish the MOS tends to improve as the number of consecutively lost packets increases, it varies from 3.58, for 1 packet lost at a time, to 4.37 for bursts of 5 consecutively lost packets; b) in French, it shows almost no variation and goes down a little bit, 4.06, 4.04, 4.04 and 3.95 for bursts of 1, 2, 3, 4 and 5 packet lost, respectively; and c) in Arabic, MOS also shows small changes, although its level is lower than that in French (see Fig. 6).

In this way, we generated the database of distorted speech signals and their corresponding quality scores. These scores are given as the average opinion of a group of human subjects, to train a neural network to *assess* speech quality. The neural network reproduces scores at the *style* of the group of human subjects that participated in the MOS tests. It must be pointed out that we included variations in network parameters that cover a wide range of combinations. Once the neural network is correctly trained it can evaluate the MOS for new network situations other than those described in Section II.

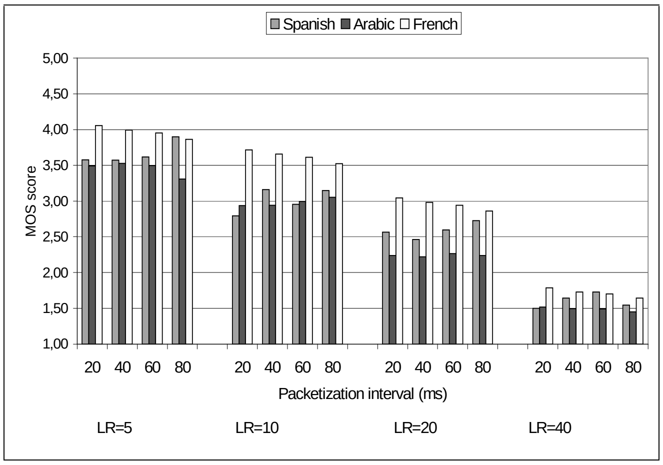

Fig. 4. MOS result as a function of packetization interval and loss rate (PCM), CL=1.

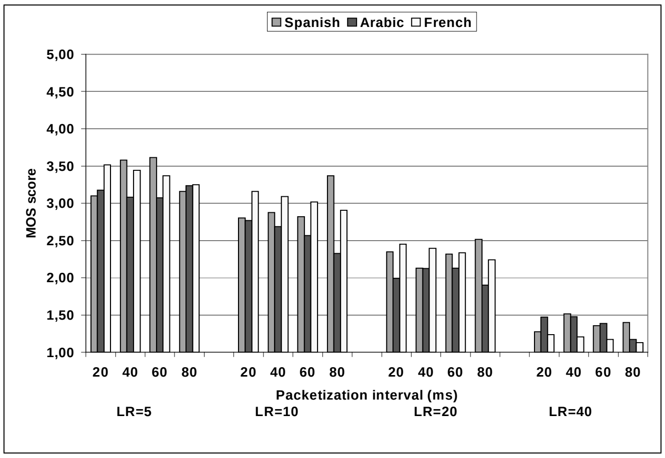

Fig. 5. MOS result as a function of packetization interval and loss rate (GSM).

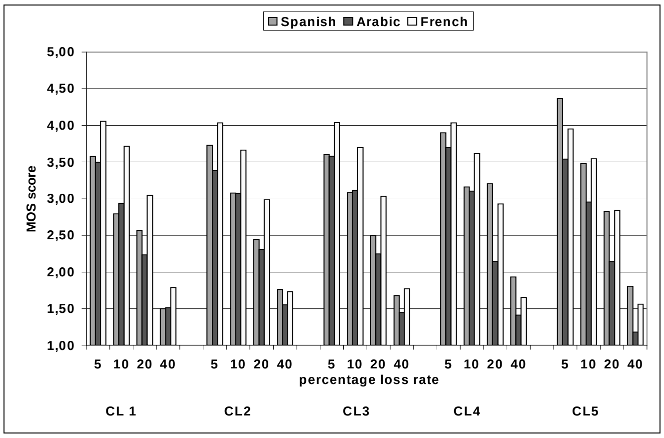

Fig. 6. Language effect on MOS for packetization interval = 20 ms and PCM.

In this paper, we only aim to show the viability of developing a neural network based automaton that may assess in real-time the quality of speech signals transmitted in a packet network. Thus, future work could include not only the implementation of our system in the environment described in Section B, but also the addition of new parameters (echo effect, two-way sessions,…) and more refined databases.

IV. Assessment of Speech Quality with Neural Networks

Based on the language dependency result obtained in Section C, when finishing the MOS tests we ended up with one set of samples for each language:

$$S_e = \{(\mathbf{x}_{e1}, q_{e1}), (\mathbf{x}_{e2}, q_{e2}), \cdots, (\mathbf{x}_{ei}, q_{ei})\},$$
$$S_a = \{(\mathbf{x}_{a1}, q_{a1}), (\mathbf{x}_{a2}, q_{a2}), \cdots, (\mathbf{x}_{ai}, q_{ai})\}, and$$
$$S_f = \{(\mathbf{x}_{f1}, q_{f1}), (\mathbf{x}_{f2}, q_{f2}), \cdots, (\mathbf{x}_{fi}, q_{fi})\}$$

Where the input vectors $\mathbf{x}_i$'s are conformed by values of network parameters (i.e., packet loss rate, packetization interval, number of consecutively lost packets) and the encoding algorithm. The speech quality scores q_i's were calculated as the average opinion of the human subjects in the MOS tests.

Thus, our neural networks based automaton has the architecture shown in Fig. 7. A three-layer feedforward neural network for each set of samples (language), and an output unit to generate a single output for the overall architecture.

A commercial simulator (Neuroshell Predictor version 2.0) was used to train the neural networks, which uses *Backpropagation* [26] as the learning algorithm. The database sizes for Arabic, French and Spanish are 100, 65 and 100 samples respectively.

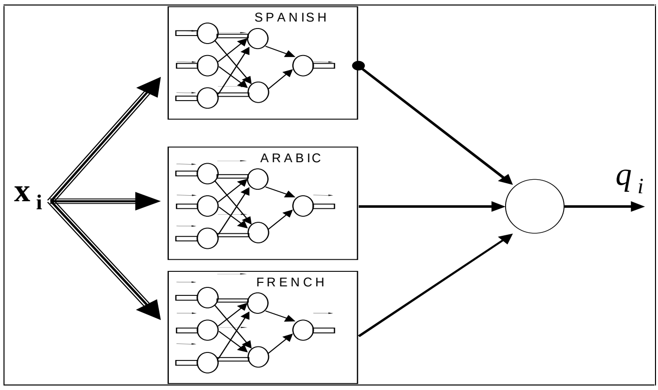

Fig. 7. Neural networks architecture.

We trained the neural network with the first 80% of database samples for each language. The remaining samples were used to test the trained neural network. By comparing the real quality scores against the neural networks predictions, the training and testing samples are plotted in Figures 8, 9 and 10. From these figures, we can see two important results. First, the neural network has the ability to learn very accurately the evaluation of the MOS for a given set of input parameters (Section III). This is clearly shown in the first 80% samples. Second, it is able to have a very precise estimation of the MOS for any new values of the input parameters. This is shown in the last 20% of the results in the curves. The statistics are:

- for Spanish: Correlation coefficient =0.998, r^2 =0.9965, and Average error =0.03;
- for Arabic: Correlation coefficient =0.994, r^2 =0.9876, and Average error =0.06; and
- for French: Correlation coefficient =0.985, r^2=0.97059, and Average error =0.10.

As can be observed, the results are very encouraging, the neural networks approach allowed us to get a very good model of a nonlinear mapping that resembles the way human subjects assess speech quality.

Regarding the updating frequency of the input parameters (loss rate, codec, packetization interval,…); if the control protocol used to carry the feedback information is RTCP, one has to update the inputs one time every 5 *secs* at the sender side. So the subjective speech quality can be calculated every 5 *secs* at the sender side. While the receiver has all the information instantaneously, all the inputs can be updated as the user prefers, but the minimum interval is the time of the packetization interval, which is 20 ms in general.

Regarding the diversity of languages in the world, it is not so difficult to carry out the MOS experiment for the majority of languages, and in this case, for the rest of languages, we can take the average of the tested languages as a references. We believe that by taking the average the accuracy of the non-supported languages will be within ± 0.5 which is tolerable in the ITU-T 5-point scale; while maintaining complete accuracy in the supported languages.

Communication networks characteristics are changing rapidly, thus neural networks adaptability is one of their greatest advantages. Neural networks can be re-trained as frequently as necessary, based on the network conditions or when different databases are gathered (based on MOS tests with different phrases, different people, or by the development of new methods for assessing speech quality, etc.). In this way, this approach allows us to ensure that speech quality assessment can be done with a nonlinear mapping that best resembles the end-users perception of speech quality signals given the current communications network situation.

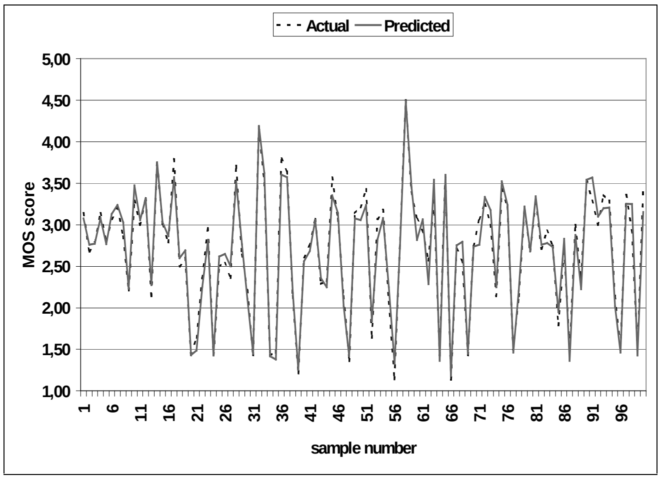

Fig. 8. Arabic Actual vs. Predicted databases.

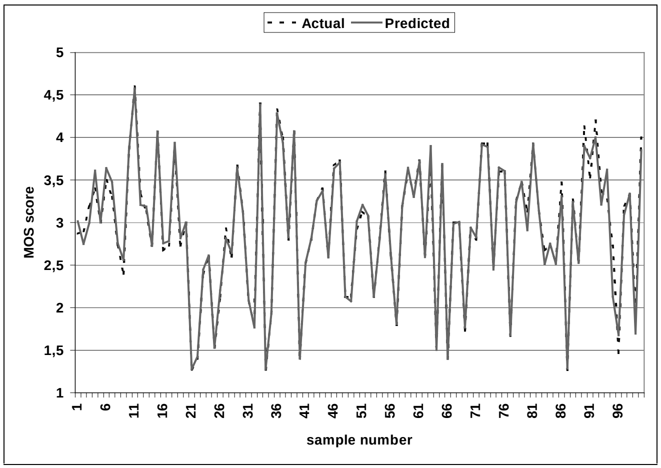

Fig. 9. Spanish Actual vs. Predicted databases.

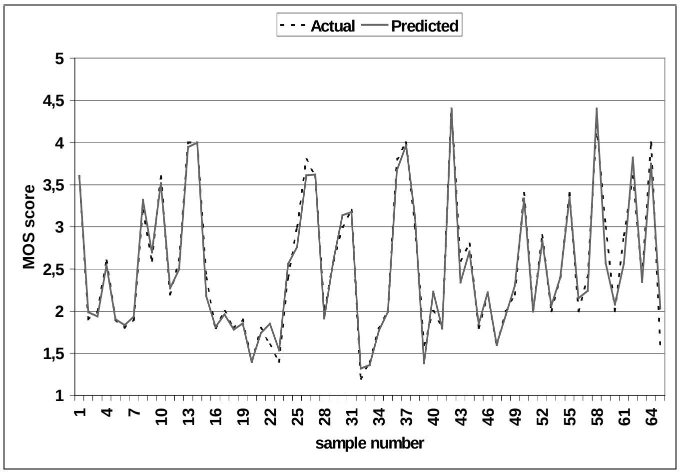

Fig. 10. French Actual vs. Predicted databases.

V. Towards a "Hybrid" Control Mechanism Integrating Network Parameters Measurements and Speech Quality Subjective Scores

It is very likely to see a progressive adaptation of most of the multimedia RTP/UDP/IP based protocols to the proposed TCP-Friendly algorithm. Although this is required to avoid the congestion collapse of the global Internet, it can drastically reduce user-perceived quality.

In [12], a TCP-Friendly Rate Control (TFRC) mechanism is proposed to control the rate of real-time applications such as audio and video conferencing.

Using TCP-Friendly like mechanisms to adapt audio applications is a way to behave as a TCP connection and hence to be less aggressive than UDP open loop flows. A TCP-Friendly sender is able to adapt its bandwidth according to network conditions. This means that it will send less data than a normal open loop UDP application. It means also that adapting the flow to the current measured network conditions will be benefic to the global Internet but may reduce dramatically the perceived quality [27]. Once the sender is obliged to decrease its bandwidth, it has to take a decision on how to decrease it. This is the goal of our control mechanism.

We propose to use the neural networks that we presented in the previous sections in order to decide on the best adaptation a sender could use when it receives congestion control information from the network. We show that the usage of the neural networks will help saving additional bandwidth in the sender and will help in improving the MOS results under the same network conditions. As shown in Section III, the same MOS scores can be obtained by several combinations of input parameters. For example, for a given bandwidth one can choose the most suitable codec that gives the best quality. On the other hand, for a given MOS score one can choose the best input parameters to reduce the required bandwidth. By changing the packetization interval (and hence changing the packet rate sent to the network), one may improve the loss ratio. Reducing the sending rate by choosing a different codec (smaller packet size), adding or removing FEC, changing the packetization interval are decisions that can change the MOS in the receiver under the same congestion circumstances and may reduce the bandwidth required by the application to deliver the real-time speech signals from the sender to the receiver.

The new control mechanism, shaded parts in Fig. 11, is composed of three parts:

1. The first part is a TFRC that periodically calculates the suggested sending rate. It receives feedback from the receiver using, in our case, RTCP extensions. The periodicity of the calculation can depend on RTCP recommendations (5 seconds and not more than 5% of the data traffic) or can be like the TFRC implementation (every Round Trip Time).
2. The second part is the neural network trained at the receiver side to measure in real time the MOS based on the network conditions. It sends feedback information to the sender. When RTCP is sent back with network statistics, we use an additional field to code the MOS result evaluated in the receiver.
3. The third part takes these results and based on internal rules, decides on the new parameters to be used. The internal rules for the moment are very simple and simply change the codec. Additional complexity can be added to integrate more parameters and to decide on the frequency of the switches. A compromise between parameters stability and bandwidth utilization should be calculated.

In order to test this control mechanism, we used an RTP/RTCP stack (the implementation goal is only for the validation of the new concept) that we upgraded with the Equation-Based Congestion Control implementation (TFRC)[16]. The network topology prepared for the tests is composed of a sender implementing TCP-Friendly, RTP and RTCP protocol necessary for the feedback information.

We used only one parameter (the codec type) to demonstrate the advantage of the new control mechanism. Adding other parameters can improve even more the MOS result and will generally depend on the nature of the application.

A local development (*bypass* software developed at INRIA Sophia Antipolis) in the router simulates a bottleneck that happens after a minute of normal operation. In addition to the bottleneck, a delay is applied to packets going through the router.

In Fig. 12, we plot the behavior of the calculated bandwidth as suggested by TFRC and the saved bandwidth when the receiver decides to change from PCM (64Kbps) to GSM (13.2Kbps) codec based on a decision taken by the sender to maintain a good quality. We can see that when the bottleneck is detected, the sender adapts its rate to even less than the available bandwidth.

In Fig. 13, we compare the two measured MOS at the receiver side for the case when the sender follows our control mechanism by changing the codec and the case when the sender blindly follows what TFRC suggests and hence continues using the PCM codec in the bottleneck periods. Clearly, the MOS is improved even in presence of congestion. The decision to change the codec, that is usually taken manually by an expert user (he has to understand all "advanced features" of the conferencing software) when he/she starts to suffer from the network conditions is taken automatically by our new controller.

Moreover, our control mechanism may also be helpful in the absence of congestion by deciding the best combination of the parameters by which the end-user will receive the best quality.

Finally, by using this control mechanism, and to maintain a certain level of quality, the sender can decide to use the exact bandwidth required by the application in the absence of congestion and not to give only an upper bound. This is shown in the first part in Fig. 12.

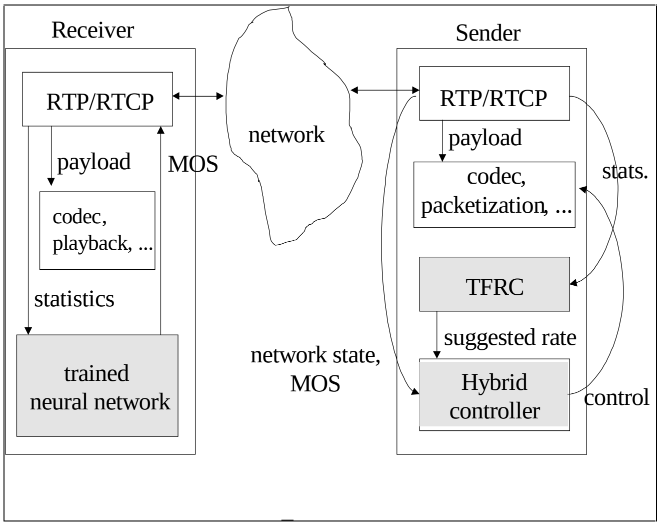

Fig. 11. Architecture of the proposed hybrid control mechanism.

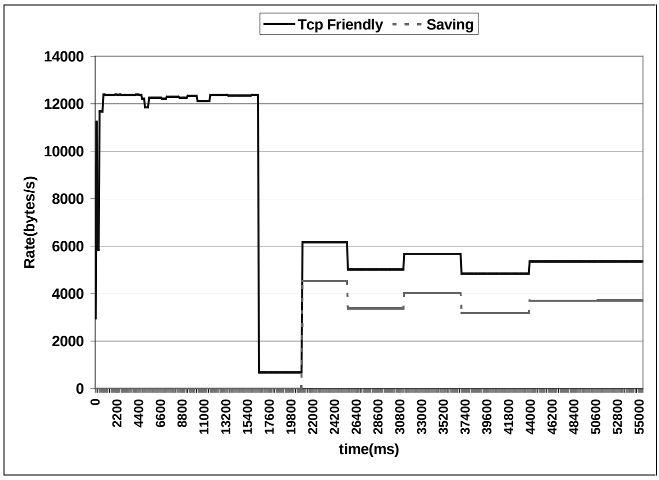

Fig. 12. Rates suggested by tcp-friendly and the saving using control rules.

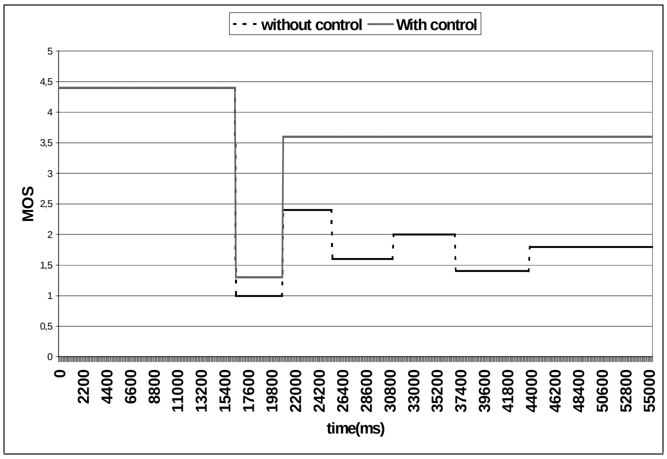

Fig. 13. MOS values with and without our control.

VI. Conclusions and Future Work

In this paper, we have presented a new mechanism that combines an automated real-time subjective speech quality assessment with a TCP-Friendly rate controller. It helps in delivering the best speech quality and in saving any superfluous bandwidth for a given network situation. The controller decides, based on the subjective quality and on the network conditions (TCP-Friendly rate controller suggestions), which parameters that should be modified to achieve this task.

We use the neural network to learn the human evaluation of the perceived quality of degraded speech signals when transmitted on the network. We take into consideration the effect of language, codec type, packetization interval, loss rate, and loss distribution.

We showed that the neural network reproduces very accurately the subjective evaluations. It is also capable of evaluating the subjective quality in presence of new sets of network parameters' values. In this way, one can automatically measure in real time and with good confidence the subjective speech quality.

In order to build the database used for training and testing the neural network, we carried out a series of field tests. This helped us in the choice of the mentioned parameters and their ranges.

The future trends may be in the enhancement of the controller by taking into account all the possible parameters. We suggest to investigate new parameters for the network, the language, the session type (one way, two way and conferences), the echo, etc.

We focused the study on speech whereas the overall method could be applied to audio and video when transmitted in real time over any kind of packet-switched network.

ACKNOWLEDGMENTS

F. Cervantes-Pérez acknowledges partial support from CONACYT under contract 993316.

REFERENCES

[1] Babich, F. and Vitez, M. “A Novel Wide-Bande Audio Transmission Scheme over the Internet With a Smooth Quality Degradation”, *ACM SIGCOMM Computer Communication Review*, vol. 30, no. 1, Jan. 2000.

[2] Bolot, J-C., Fosse-Parisis, S., Towsley , D. “Adaptive FEC-Based error control for Internet Telephony*”, Proceedings of Infocom '99,* New York, NY, March 1999.

[3] Bolot, J-C., Garcia, A.V., “The case for FEC-based error control for packet audio in the Internet”, ACM Multimedia Systems, 1996

[4] Bolot, J-C. “End-to-end packet delay and loss behavior in the Internet”, *Proceeding of ACM Sigcomm '93*, pp. 289-298, San Fransisco, CA, Sept. 93.

[5] Clark, D. and Tennenhouse, D. “ Architectural Considerations for a New Generation of Protocols”, *ACM SigComm Symposium*, September, 1990

[6] Cray, A. “Voice over IP: Hear's how” *Data Communications International*, vol. 27, no. 5, pp. 44-59, Apr. 1998.

[7] De Vleeschauwer, D., Janssen, J., Petit, G. H. “Delay bounds for low bit rate voice transport over IP networks”. *Proceedings of the SPIE Conference on Performance and Control of Network Systems III*, Vol. 3841, pp. 40-48, Boston (MA), 20-21 Sept. 1999.

[8] De Vleeschauwer, D., Petit, G.H., Steyaert, B., Wittevrongel, S. and Bruneel, H. “An Accurate Closed-Form Formula to Calculate the Dejittering Delay in Packetised Voice Transport”. *Proceedings of the IFIP-TC6/European Commission International Conference Networking 2000*, PP. 374-385, Paris, France, 14-19 May 2000.

[9] Deller, J., Proakis, J., Hansen, J.H.L., *Discrete-Time processing of speech signals*. New York, MacMillan Series for Prentice Hall, 1993.

[10] Dimolitsas, S. “Objective speech distortion measures and their relevance to speech quality assessments”, *IEEE Proceedings*, Vol. 136, Pt.I, No. 5, October 1989.

[11] Floyd, S., Jacobson, V., Liu, C., McCanne, S., and Zhang, L. “A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing”, *IEEE/ACM Transactions on Networking*, December 1997, V. 5, N. 6, pp. 784-803.

[12] Floyd, S., Handly, M. Padhye, J. and Widmer, J. “Equation-Based Congestion Control for Unicast Applications”, *SIGCOMM 2000*, May 2000.

[13] Hansen, J.H. and Pellom, B.L. “Speech Enhancement and Quality Assessment: A Survey”*, submitted to IEEE Sig. Proc. Mag.* Nov. 1998.

[14] Hassan, M., Nayandoro, A., Atiquzzaman, M. “Internet Telephony: Services, Technical Challenges, and Products”, *IEEE Communications Magazine*, pp. 96-103, April 2000.

[15] http://ucserv.seas.smu.edu/ee6392-30/index.html.

[16] http://www.aciri.org/tfrc/code

[17] ITU-T Recommendation P.800, “Methods for subjective determination of transmission quality”.

[18] Janssen, J., De Vleeschauwer, D., Petit, G., “Delay and Distortion Bounds for Packetized Voice Calls of Traditional PSTN Quality”, *Proceeding of IPTEL 2000*, Berlin, Apr. 2000.

[19] Kirby, D., Warren, K., Watanabe, K. “Report on the Formal Subjective Listening Tests of MPEG-2 NBC multichannel audio coding”, ISO/IEC JTC1/SC29/WG11, N1419, Nov. 1996.

[20] Kostas, J., Borella, M., Sadhu, I., Schuster, G., Grabiec, J. and Mahler, J. “Real-time Voice over Packet Switched Networks”, *IEEE Network*, Jan./Feb. 1998, vol. 12, no. 1, pp. 18-27.

[21] Mohamed, S., Cervantes, F.P., Afifi, H. ” Audio Quality Assessment in Packet Networks: an "Inter-Subjective" Neural Network Model” *Proc. International Conference on Information Networks*, Japan, Jan. 2001.

[22] Ojala, P., Toukomaa, H., Moriya, T. and Kunz, O., “Report on the MPEG-4 speech codec verification tests”, ISO/IEC JTC1/SC29/WG11, MPEG98/N2424, Oct. 1998.

[23] Perkins, C., Hodson, O. and Hardman, V. “A Survey of Packet-Loss Recovery for Streaming Audio”, *IEEE Network*, Sept./Oct. 1998, vol. 12, no 5., pp. 40-48.

[24] Quackenbush, S.R., Barnwell, T.P. and Clements, M.A. *Objective measures of speech quality*. New Jersey, Prentice Hall, 1988.

[25] J. Rosenberg, L. Qiu and H. Schulzrinne “Integrating Packet FEC into Adaptive Voice Playout Buffer Algorithms on the Internet”, IEEE Infocom 2000, March 2000.

[26] Rumelhart, D.E., Hinton, G.E., and Williams, R.J. “Learning internal representations by error propagation”. In: *Parallel Distributed Processing*, vol. 1. Cambridge, Massachusetts, MIT Press, 1986.

[27] Tan, D. Zakhor A. “Real-time Internet Video using Error Resilient Scalable Compression and TCP-Friendly Transport Protocol”. *IEEE Transactions on Multimedia*, May 1999.

[28] Watson, A. and Sasse, M.A. “Measuring Perceived Quality of Speech and Vidio in Multimedia Conferencing Applications”. Proceedings of ACM Multimedi’98, pp 55-60, 12-16 Sept. , 1998.

[29] Watson, A. and Sasse, M.A. “Evaluating Audio and Video Quality in Low-Cost Multimedia Conferencing Systems”, *Interacting with Computers*, 8(3):255-275, 1996.

[30] Wright, D.J. “Voice over ATM: An Evaluation of Implementation Alternatives”, *IEEE Communications Magazine*, vol. 34, no. 5, pp. 72-81, 1996.