Real-Time Caption Streaming over WiFi Network

D. Maniezzo*†, M. Cesana†, P. Bergamo†, M. Gerla*, K. Yao†

*Computer Science Department † Electrical Engineering Department
University of California Los Angeles - UCLA
Los Angeles, CA 900095-1594, USA
maniezzo@ee.ucla.edu, cesana@cs.ucla.edu, pbergamo@ee.ucla.edu, gerla@cs.ucla.edu, yao@ee.ucla.edu

***Abstract*— The use of laptop and handheld computers in educational environments has changed the nature of teaching, introducing new ways for students to interact with materials, teachers, and their classmates. Technological advances allow handheld devices to be equipped with faster processors and wireless interfaces, making the performance comparable to laptop computers. In this paper we propose a simple but effective scheme according to which each student can visualize in real-time and store the captions of the ongoing lecture. The system is based on IEEE802.11b Multicast protocols and implements a redundant transmission mechanism to mitigate the errors due to the unreliable wireless channel.**

I. Introduction

Technological advances in computing will make classrooms and laboratories much more accessible and effective. A new generation of high technology classrooms is becoming a necessity on college campuses, and most teaching institutions recognize the need to incorporate computers connected to the Internet into their students' educational experience. Handheld devices have been successfully used to encourage student collaboration and interaction in the classroom [1] [2] [3] [4] [5].

The rapidly decreasing price of handheld computers combined with wireless connectivity can boost the spreading of technology within the educational process. Students can be easily given access to much richer and real-time information useful for their own learning process. For example, they can access distributed resources such as electronic libraries, internet databases, personal file, database collections, real-time data and laboratory equipment outside the classroom, all while in a lecture, discussion, or group meeting. Mobile handheld or laptop computers with wireless connectivity to the Internet introduce the opportunity of mobility for students and the challenge of effectively using the tools for educators.

Furthermore, technology can be extremely useful in those situations where a particular support to the education process is needed. As an example, video support during classes can help people with hearing impairments.

In this paper, we propose a simple but effective scheme based on the IEEE802.11 technology according to which each student can visualize in real-time and store the captions of the ongoing lesson (Figure 1). Such a solution can be useful in many real situations: to help the comprehension of foreign students, students with hearing impairment, or just as a support to notes taking during classes (students can focus on understanding the lecture since notes are taken by the electronic devices). Furthermore, our proposed scheme can be integrated with the lecture slides, and a synchronized system having slides and associated caption can be implemented.

A similar application, named Multicast Power Point (MPPT), has been recently proposed by Microsoft [6]. The biggest difference between our solution and MPPT is that in our case the information is transmitted in pure multicast manner with one active sender and several passive receivers. In MPPT, on the contrary, some kind of signalling information exchange is required from receivers to sender. For this reason, it is not well suited for implementation in a wireless broadcast scenario based on IEEE802.11 where multiple collisions at the sender side can happen if the receivers have to acknowledge the sender transmissions.

In this work, we focus on the caption streaming system leaving the slide transfer to future studies. Key points of our solution are:

- Speech recognition algorithm running in the central host,
- Multicast transmission via IEEE802.11b from the central host (sender) to the devices (receivers);

We have implemented our solution in a testbed through which we are able to obtain some performance evaluation.

The paper is organized in the following way: in section II we provide a detailed description of the basis of the proposed solution, in section III we describe the testbed we implemented and give some performance evaluation. Finally, section IV contains some concluding remarks.

II. Wireless Caption Delivery Scheme

A. Basic Idea and Protocol Description

Wireless multimedia can be a powerful support to education. Today, almost every student owns a portable device whether it is a laptop or a PDA (Personal Digital Assistant). The general trend of many institutions is to introduce technology in their classes as an effective support to the learning process [7] [8] [9].

We propose a scheme which can easily be implemented in any classroom scenario. We will name our scheme Wireless Captions Delivery (WCD). In our scenario, each student has a IEEE802.11b wireless PCMCIA card to be plugged either in a laptop or in a PDA. The instructor is equipped with a headset and an IEEE802.11b enabled station, running a speech

0-7803-7724-9/03/$17.00

Fig. 1. Caption Streaming over WiFi Networks

detection software. The voice of the lecturer is captured by the headset, converted into a text file by the speech recognition software and sent via a IEEE802.11b multicast transmission to all the students. In the following, we present in details the proposed solution and its features.

In this paper, we focus on the transmission part of our solution and we make simplifying assumptions on the speech recognition part.

The lecturer's host runs a client program which handles the multicast transmission and is wirelessly connected to the receiving stations that run a server program able to receive the packets and display the speech in a real-time fashion.

The choice of multicast protocol is simply due to the point-to-multipoint nature of transmission. To understand our choice, let us consider the usual communication scheme. In order to deliver captions to students' terminals, a number of unicast sockets (peer to peer reliable transport layer communications) equal to the number of students could be used. Since captions are to be sent over each socket, the wasted bandwidth is very large. The actual problem is that such a scheme is not scalable and it could generate a dramatic congestion in situations of very noisy wireless channel.

In the proposed scheme, there is a sender (the lecturer's host) and many passive receivers. Basically, a simple broadcast transmission over the wireless channel could be used for this purpose. Multicast is essentially based on a broadcast transmission but it offers some additional features. We have chosen Multicast because we consider to have students also in other classrooms (remote-lecture with some video-hardware). The caption-streaming to other classrooms can be simply realized through the wireline network of the department. A wireless (multicast enabled) access point connected to the core-network shall be in the lecturer's classroom. The same shall be in the other classrooms. The portable devices in other classrooms simply have to join the right multicast group. We consider also to have some backup host connected to the wireline network. The backup host could be some department server with the goal of recording the lecture.

Multicast is implemented on the unreliable UDP as transport layer since a connection-oriented reliable communication is hard to realize in point-to-multipoint scenarios, as briefly discussed. Since we are working in a wireless scenario some transmission from the sender to the receivers can get lost due to a temporary channel failure. In this case two possible approaches can be followed: first, trying to recover the lost information at the receiving side with some kind of semantic aware software; second, devise a mechanism robust to channels errors. Since the receivers in this particular scenario have to run on relatively small devices, like PDAs, we chose the second approach. In this situation, two solutions can be applied: introducing some redundancy in the multicast information to remove retransmissions, or implementing a retransmission request protocol.

Multicast transmissions are sent in a broadcast way by IEEE802.11b DCF protocol, which is a CSMA based protocol. If students' terminals start transmitting some packets (such as retransmission requests), broadcast packets can be lost due to collisions. Following IEEE802.11b protocol, a unicast packet lost for collision shall be retransmitted (since no ack is received by the transmitter). Unfortunately, there is no way to know whether the packet is lost in a broadcast transmission or not (collision detection as in IEEE802.3 cannot be used in wireless channels).

Thus, in this work we chose to adopt the redundancy solution which is much more suited for a broadcast environment where handling retransmissions and retransmissions requests can be cumbersome. For this reason we implemented the Redundant Transmission (RT) protocol. We will show with results of a testebed that the simple idea of redundancy is effective for such a scenario.

In details, each multicast packet contains N **fragments** (i.e., a subset of subsequent words) of the generated text and the choice of which fragments have to be transmitted within a packet is based on a sliding window algorithm. Figure 2 shows the algorithm in the case of $N = 4$. The first transmitted packet brings the fragments numbered from P to $P + 3$. The next transmission delivers the fragment numbered from $P + 1$ to $P + 4$, and so forth. In this way, even if some broadcast packets are lost, the receiving terminals can reconstruct the information of the lost packet by using the redundancy in transmission. Obviously, the greater the N, the greater is the system robustness to channel errors.

On the other hand, an increase of N increases the length of the 802.11b broadcast packets and, consequently, decreases the available bandwidth. Furthermore, the longer the broadcast packet, the higher the probability of loosing them due to channel errors, and an increase in N ends up in increasing the fragment delivery delay. Thus, a trade off choice between the error protection level and the delivery delay must be taken.

At the end of the lecture, without real-time constraints, the receivers (if willing) can require **the retransmission of the**

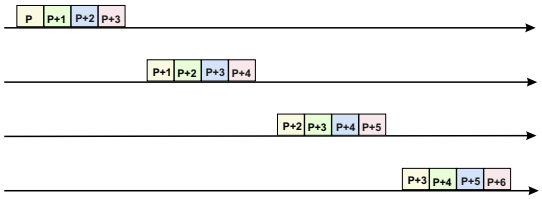

Fig. 2. Redundant transmission scheme example. The Sliding Window Mechanism

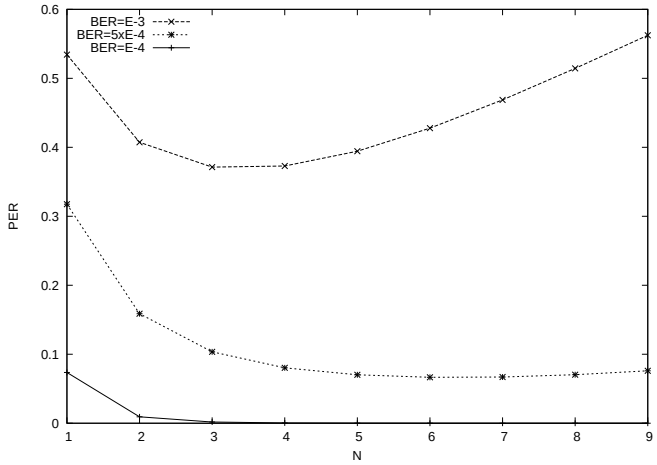

Fig. 3. Fragment Error Probability as a function of N (analytical model).

missed fragments via a reliable transport protocol (TCP).

B. Some Analytical Considerations

Let us consider a very simple analytical model which can give some insights on the proposed scheme. For the sake of simplicity, we assume channel errors are independent bit by bit, and no channel coding mechanisms exist.

Given a fixed value of bit error probability (BER), the probability of a fragment to be lost ($FragmentErrorRate - FER$) is given by the following expression

$$FER = \sum_{i=N}^{\infty} (1 - (1 - BER)^L)^i,$$

where N is the number of fragments for each broadcast packet and L is the average length in bits of the broadcast packet itself. L is computed according to the following equation

$$L = NL_{frag} + H,$$

where L_{frag} is the average length of a fragment and H takes into account all the protocol headers. Figure 3 shows the FER versus N for different values of the bit error probability BER (H=64 bytes). In the figure, we assumed the approximation $FER = (1 - (1 - BER)^L)^N$.

This simple analytical model assures that there is an optimal value of N in different conditions. We have investigated the performance of the system as a function of N in a real testbed. In the next section, we present a detailed description of the testbed we have implemented.

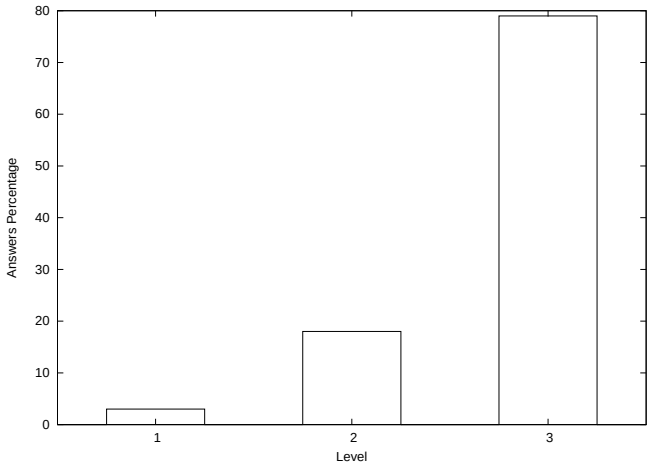

Fig. 4. Experienced Quality.

III. TESTBED DESCRIPTIONS

We have implemented all the transmission protocol in Java [10]. This choice assures an easy portability of the software onto many platforms. The sending side of the software basically checks continuously a file, supposed to be the output of the speech translator program. Those kinds of programs periodically write on the output device a set of words. Throughout the paper we will call these sets of words fragments, while each group of fragments sent via IEEE802.11 will be referred to as a packet. New written words are kept by our software and sent to receivers. The speech recognition software running at the sender side is an Automatic Speech Recognitor (ASR) built on the ISIP Foundation Classes. ISIP is the Institute for Signal and Information Processing at Mississippi State University. Further information can be found in [11].

On the receiver side, the text is displayed. When the sender signals that communication is over, receivers request for lost fragments. Both sender and receiver software collect statistics.

The testbed we set up is described by Figure 5. In details, we used 4 laptops running GNU/Linux Red Hat 8.0 [12] and 3 iPAQs [13] running GNU/Linux Familiar 0.6 [14]. All the mobile devices are equipped with Cisco Aironet 350 PCMCIA wireless adapters. Laptop A in Figure 5 runs the sender program which handles the IEEE802.11b broadcast transmissions and is equipped with the speech generation software. All the other devices run a receiver software which handles the reception of the packets, display the sent text and collect performance statistics.

As a first step, we tested our scheme in a real scenario, where at the sender side a lecturer was holding a presentation titled ”IEEE802.11 MAC layer Overview”, for an overall duration of 35 minutes. We assigned the receiving units to 6 people and we asked them to evaluate the effectiveness of the mechanism by assigning a grade ranging from 1 (low quality) to 3 (high quality). We repeated this experient 5 times, each time with a different speaker and a different audience.

Figure 4 shows the result on the experienced quality in terms

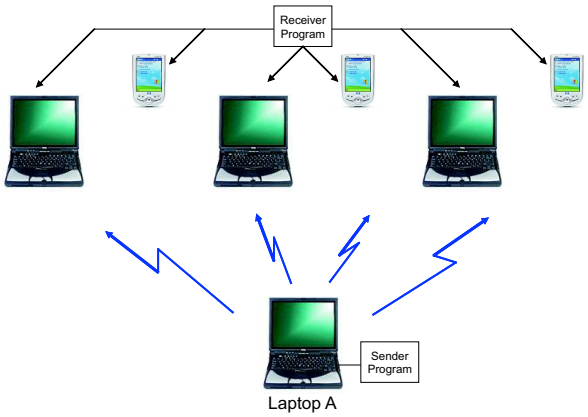

Fig. 5. Testbed description.

of percentage of people for each quality level $(1, 2, 3)$. Among the people interviewed, about 80% stated that the quality of the proposed scheme is fully satisfactory.

Since the overall quality of the scheme depends both on the speech recognition software and on the transmitting protocol, to obtain a deeper understanding of the performance of the latter one, we ran some experiments where the speech recognition process in the lecturer device was simulated. In this way, the traffic being transmitted becomes a controllable parameter, and we can focus our attention on the transmitting part of the scheme.

In order to simulate the process of speech to text translation we developed a software tool which generates words out of a fixed set and with a parametric rate. Parameters for the following analysis are derived from [15]. Sets of words are generated randomly with an inter-arrival time distributed according to a negative exponential with mean $1s$. Each set of words brings a number of words uniformly distributed between 1 and 8, and the word length is drawn from an exponential distribution with mean equal to 9 characters of 8 bytes each. The traffic parameters are summarized in Table I.

Parameter	Value
Interarrival Time	Exponential Distributed
Interarrival Time Average	$1s$
# Words per Fragment	$U[1, 8]$
# Chars per Word	Discrete Poissonian Distributed
# Chars Average per Word	9

TABLE I
TRAFFIC PARAMETERS.

For the performance evaluation, we considered four values of the parameter N, from 0 to 4. For each of these value we ran 5 ten minutes long experiments where we generated the fragments to be transmitted according to the model described in section II. The results shown in this section are obtained averaging the statistics over all the experiments runs.

Figure 6 shows the packet error rate and the fragment error rate with respect to the parameter N. The packet error rate refers to the entire broadcast packet, while the fragment error rate is defined as the probability of loosing a single set of words. Obviously, the probability of a broadcast packet to be corrupted increases with the length of the packet itself, that is with N. On the other hand, the probability that a voice fragments is lost has a minimum for $N_{min} = 1$. For low values of N ($N = 0$ or$N = 1$), increasing N increases the level of protection for a fragment, therefore the fragment loss rate starts decreasing. Above a certain value (i.e., N_{min}), the positive effect of the higher redundancy is overwhelmed by the increase in the packet error rate and, as a consequence, the fragment error rate slightly increases. The value N_{min} depends on the wireless channel conditions, and in general is proportional to the instantaneous BER. The fragment error rate is very low in the considered scenario.

Figure 7 depicts the average delivery delay of a voice fragment versus the parameter N, while in figure 8 the normalized standard deviation of the packet delivery delay versus the number N is reported. The average delay is computed assuming the propagation time is negligible. For a better understanding, the delay of a fragment is equal to zero if the fragment is correctly received the first time is transmitted. Otherwise, the delay is proportional to the number of broadcast packets to be transmitted before the fragment is correctly received. On the other hand, the normalized standard deviation is the ratio between the standard deviation of the delay distribution, and its mean value.

From Figure 7, the average delay obviously increases with the redundancy in transmission and is extremely low for all the tested values of N. On the other hand, from Figure 8, it is clear that the standard deviation of the fragment delivery delay can be much higher than its mean value. For example, if $N = 2$, the standard deviation of the delay is almost 20 times the mean value. This means that there are fragments which reach the destination 100ms after first transmission. Thus, the optimal value of N must be chosen from a trade off between the need of protecting the fragments from transmission errors and consequent retransmissions, and the need of keeping the delivery delay controlled in order to preserve the real time characteristics of the application.

To summarize the results obtained in this section we can state that:

- The choice of the redundancy parameter N is based on a trade off between redundancy protection and delivery delay. The optimal value of N depends on the wireless scenario considered;
- In the testbed we implemented, the broadcast transmission is very robust to channel errors so a low value of the redundancy parameter is required, typically at $N = 1$.

IV. CONCLUSIONS AND FUTURE WORK

Computers and transmission technology can provide a substantial help to the learning process. Many universities are actively working to introduce technology in their classroom in terms of laptops, PDAs and mobile devices equipped with some kind of wireless technology (Bluetooth, IEEE802.11,

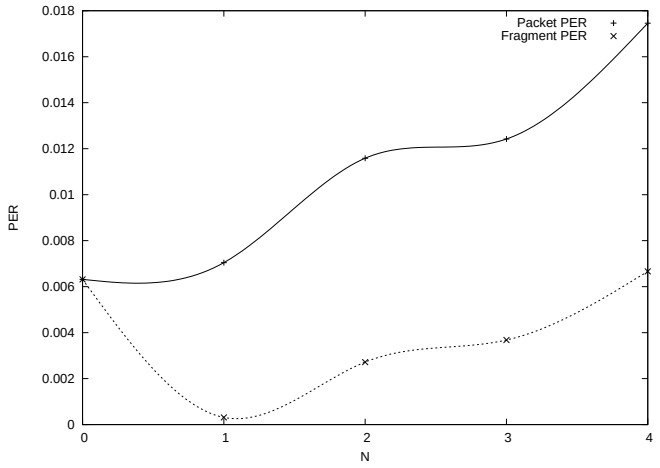

Fig. 6. Packet Error Probability and Fragment Error Probability versus the parameter N.

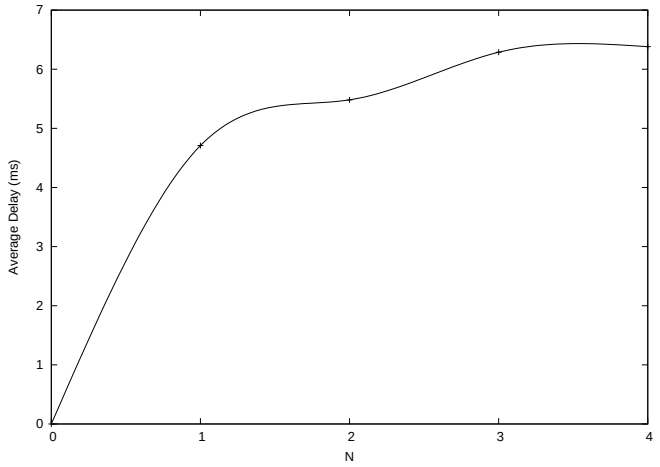

Fig. 7. Average Packet Delivery Delay versus the parameter N.

etc.). In this paper we proposed a novel solution to broadcast the captions of an ongoing lecture to all the students in a real time manner. Our scheme, named Wireless Caption Delivery (WCD), exploits a voice recognitor to capture the instructor's speech which is transferred to the students via a IEEE802.11b broadcast transmission. WCD can provide a good support to basic lesson comprehension and can be helpful to foreign students or people with hearing impairments. We have set up a testbed to evaluate the proposed scheme and we have presented some preliminary performance measures.

As a next step of our work, we are planning to test our scheme under different propagation and mobility conditions. Moreover, we are studying the possibility of enhancing the proposed solution with other features, such as a synchronized mechanism between streaming captions and the projected slides.

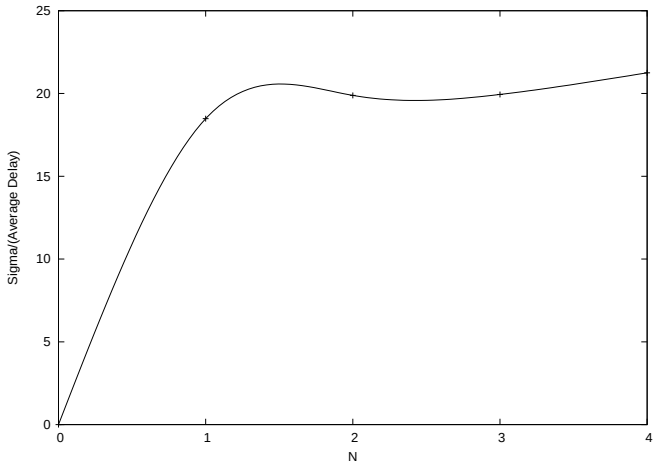

Fig. 8. Normalized Standard Derivation of the Packet Delivery Delay versus the parameter N.

V. Acknowledgements

This work has been partially funded by the HP Mobility Grant year 2002 and the UC Core program Core01-10091 under the sponsorship of ST Microelectronics. Corresponding Author: D. Maniezzo UCLA Department of Computer Science, BH 3731, 420 Westwood Plaza, Los Angeles, CA 90024 - USA, dmaniezzo@ieee.org.

References

[1] M. J. Jipping, J. Krikke, S. Dieter, and S. Sandro, "Using handheld computers in the classroom: Laboratories and collaboration on handheld machines", in Proceedings of the ACM SIGCSE, 2001.
[2] R. McFall and G, Stegink, "Introductory computer science for general education: Laboratories, textbooks, and the internet", in SIGCSE Bulletin, March 1997.
[3] W. Burleson, A. Ganz, and I. Harris, "Educational innovations in multimedia systems", in Proceedings of ASEE/IEEE Frontiers in Education, 1999.
[4] P. G. Shotsberger and R. Vetter, "Teaching and learning in the wireless classroom", IEEE Computer, vol. 34, no. 3, March 2001.
[5] R. P. Jones, F. Ruehr, and R. Salter, "Web-Based laboratories in the introductory curriculum enhance formal methods", in SIGCSE Bulletin, March 1996.
[6] http://research.microsoft.com/barc/mbone/mppt.asp
[7] "UNCW Numina Project", University of North Carolina at Wilmington, http://aa.uncwil.edu/numina/.
[8] "Wireless Classroom Project", Department of Computer Science Department University of Kentucky, http://www.dcs.uky.edu/∼wc/rframe.html.
[9] "Ubiquitous Instruction using Mobile Internet Access via an engineering campus Mobility Network", Computer Science Department, University of California Los Angeles, http://www.cs.ucla.edu/NRL/hp_project/index.htm.
[10] http://java.sun.com
[11] http://www.isip.msstate.edu/project/speech/
[12] http://www.redhat.com
[13] http://www.hp.com
[14] http://familiar.handhelds.org
[15] H. Nanjo, T. Kawahara, "Speaking-Rate dependent Decoding and Adaptation for spontaneous lecture speech recognition", ICASSP 2002