Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Networks

Abdelbasset Trad[1], Qiang Ni[1], and Hossam Afifi[2]

[1] INRIA, Planete Project
2004 Route des Lucioles, BP-93
06902 Sophia-Antipolis, France
{atrad, qni}@sophia.inria.fr
[2] INT-INRIA
9 rue Charles Fourier
91011 Evry, France
{Hossam.Afifi}@int-evry.fr

Abstract. In this paper, we present an adaptive architecture for the transport of VoIP traffic over heterogeneous wired/wireless Internet environments. This architecture uses a VoIP gateway associated with an 802.11e QoS enhanced access point (QAP) to transcode voice flows before their transmissions over the wireless channel. The instantaneous bit rate is determined by a control mechanism based on the estimation of channel congestion state. Our mechanism dynamically adapts audio codec bit rate using a congestion avoidance technique so as to preserve acceptable levels of quality. A case study presenting the results relative to an adaptive system transmitting at bit rates typical of G.711 PCM (64 kbit/s) and G.726 ADPCM (40, 32, 24 and 16 kbit/s) speech coding standards illustrates the performance of the proposed framework. We perform extensive simulations to compare the performance between our adaptive audio rate control and TFRC mechanism. The results show that the proposed mechanism achieves better voice transmission performance, especially when the number of stations is fairly large.

1 Introduction

The Internet heterogeneity is increasing due to the fast deployment of wireless local area networks (WLANs). WLAN hold the promise of providing unprecedented mobility, flexibility and scalability than its wired counterpart. At the same time it seems inevitable that future telephony services will be based on IP-technology. There is a serious concern from the operators side as to offer at least the current “circuit switched” quality for future voice over IP (VoIP) communications. In order for this to become reality, a lot of issues related to VoIP transmission over heterogeneous wired/wireless networks must be solved. In WLAN environments bandwidth is scarce and channel conditions are varying and highly lossy. Even if a lot of voice codec can tolerate some small loss without severe degradation, voice traffic has unacceptable performance if long delays are incurred. Moreover, the original IEEE 802.11 WLAN standard [1] has been mainly designed for data applications. Two different channel access mechanisms are specified in the 802.11 standard, namely, the contention-based DCF and the polling-based PCF access mechanisms. While DCF and PCF may provide satisfactory performance in delivering best-effort traffic, they lack the support for QoS requirements posed by real time traffic such as VoIP. These requirements make the DCF scheme an infeasible option to support QoS for VoIP traffic. Furthermore, apart from these limitations, a typical WLAN with 11Mbps bandwidth could only support a very limited VoIP connections in DCF mode. On the other hand, PCF mode, with a centralized controller, represent another promising alternative to providing QoS in WLAN [2]. However, studies on carrying VoIP over WLAN in PCF mode in [3] found that when the number of stations in a basic service set (BSS) is large, the polling overhead is high and results in excessive end-to-end delay and that VoIP still get poor performance under heavy load conditions. The medium access control (MAC) layer of the emerging IEEE 802.11e [4] standard tries to support QoS in 802.11 wireless networks using a new Hybrid Coordination Function (HCF) that provides stations with prioritized and parameterized QoS access to the wireless medium. The simple HCF scheduler proposed in 802.11e standard considers the QoS requirements of flows by allocating transmission time to stations based on their mean sending rate and mean frame size. In this work, we investigate the performance limitations in the case of a large number of VoIP flows transmitted over an IEEE 802.11e WLAN. We specifically address the problem of long distance VoIP transport over heterogeneous wired/wireless networks. In the studied case we consider VoIP traffic transmitted from a wired Internet part through a last-hop wireless link that represents the bandwidth bottleneck. All voice traffic needs to be routed through an 802.11e QAP (QoS-enhanced Access Point). The QAP becomes heavily loaded, especially when the number of active stations is fairly large and this results in different types of audio performance degradation (loss due to congestion, loss due to bit errors at the link layer and packet delay). We show through simulations the performance of VoIP according to the number of wireless stations in a BSS. We propose an architecture that is based on a VoIP gateway for interworking the wired and wireless networks. The gateway communicates with a QAP in order to adapt coding rate of voice flows according to the radio channel conditions. Simulations show that our adaptive audio rate control outperforms TFRC mechanism. The paper is organized as follows: Section 2 describes related work on rate and loss control for multimedia applications. Section 3 states the problem. In Section 4, we advance the proposed architecture and we explain our adaptation algorithm. We show simulation results in section 5. Finally, Section 6 concludes the paper.

2 Rate and Loss Control for Multimedia Applications

Rate control is an important issue for both wired and wireless multimedia applications using unresponsive transport protocols (i.e., UDP and RTP). A proper form of congestion control is needed in order for these applications to share congested bandwidth fairly with each other and with TCP-based applications. Many schemes were developed based on TCP-Friendly control mechanisms. These mechanisms can be classified into three main categories: equation-based mechanisms, window-based mechanisms and additive increase, multiplicative decrease (AIMD) mechanisms. Equation-based rate control [9][11] is a widely popular rate control scheme over wired networks, also known as TCP-Friendly Rate Control (TFRC). In this scheme, the sender uses an equation characterizing the allowed sending rate of a TCP connection as a function of the RTT and packet loss rate, and adjusts its sending rate according to those measured parameters. A key issue is than to choose a reliable characterization of TCP throughput. A formulation of the TCP response function was derived in [10], it states that the average throughput of a TCP connection is given by:

$$T(Bytes/sec) = \frac{s}{t_{RTT}\sqrt{\frac{2l}{3}} + t_{RTO}(3\sqrt{\frac{3l}{8}})l(1+32l^2)} \tag{1}$$

Equation (1) roughly describes TCP's sending rate as a function of the frequency of loss indication l, round trip time t_{RTT} and packet size s. This equation reflects TCP's retransmit timeout behavior, as this dominates TCP throughput at higher loss rates. In the scheme proposed in [10] the receiver acknowledges each packet, and at fixed time intervals the sender estimates the packet loss rate experienced during the previous interval and updates the sending rate using equation (1). This scheme updates the sending rate at fixed time intervals, hence it is suitable for use with multimedia applications. Nevertheless it has the disadvantage of having a poor transient response at small time-scales [16]. In [11], Floyd et al. developed the TFRC protocol. TFRC estimates the recent *loss event* rate of a connection at the receiver. A *loss event* consists of one or more packets dropped within a single RTT. The algorithm used for calculating the loss event rate (average loss interval) offers a good responsiveness to changes in congestion while avoiding abrupt reductions of the sending rate in response to a single loss. To behave in a TCP-friendly manner, the sender adapts according to an equation that models TCP response function in steady-state. The main advantages of TFRC are: first it does not cause network instability, thus avoiding congestion collapse. Second, it is fair to TCP flows. Third, the TFRC's rate fluctuation is significantly lower than that of the standard TCP congestion control algorithm, making it more appropriate for real-time applications that require a smooth congestion control and a constant quality. Window-based mechanisms such as TEAR [20] maintain a congestion window to control the transmission of packets. TEAR shifts TCP emulation to the receiver and uses a sliding window to smooth sending rates. The main disadvantage of this type of mechanisms is the lack of flexibility related to the TCP window dynamics [16]. Unlike window-based mechanisms, AIMD mechanisms [18][19] are rate-based congestion control mechanisms that are not applied to a congestion window. The Rate Adaptation Protocol [18] implements an AIMD algorithm based on regular acknowledgments sent by the receiver. In [19], the authors propose an end-to-end rate adaptation scheme that adjusts the transmission rate of multimedia applications to the congestion level of the network. Based on the estimation of the loss rate and the RTT obtained from the regular information of RTCP [8] reports, the sender increases the transmission rate during underload periods and reduces this rate during congestion periods, while avoiding an aggressive adaptation behavior. Although TCP-friendly rate control mechanisms provide relatively smooth congestion control for real-time traffic, they are more appropriate for use over wired IP networks. For multimedia applications over wireless, packets can be corrupted by wireless channel errors at the physical layer and thus TFRC cannot distinguish between packet loss due to congestion and that due to bit errors. TFRC, designed to deal with congestion in wired networks, treats any loss as a sign of congestion. End-to-end statistics can be used to help detecting congestion when packet loss happens. For example, by examining trends in the one-way delay variation, loss could be interpreted as a sign of congestion if this delay is increasing, and as a sign of wireless channel error otherwise [14]. The scheme presented in [13] combines packet inter-arrival times and relative one-way delay to differentiate between congestion and wireless packet losses. This scheme is based on the observation that the one-way delay increases monotically when there is congestion and that the inter-arrival time is expected to increase if there is wireless channel packet loss. Loss differentiation algorithms can then be combined with TFRC to achieve a rate control over heterogeneous Internet environments. The second limitation of TFRC mechanisms is that they are originally designed for applications that use fixed packet size, and vary their sending rate in packets per second in response to congestion. Hence, they should not be used for applications that vary their packet size instead of their packet rate in response to congestion [12]. Varying the packet size during the time interval between two estimations of the sending rate distorts packet-based measurement of loss event. In some situations, using rate control alone does not solve the performance degradation. Such situations may be short-term transient congestion, congestion caused by others' traffic or residual bit errors caused by a noisy wireless link. Forward Error Correction (FEC) has been one of the main methods used to protect against packet loss over packet switched networks and also to improve the quality of noisy transmission wireless links. The amount of FEC information should be tuned according to the characteristics of the loss process in order not to increase bandwidth requirement (and hence the packet loss rate) when the channel is loss free and also not to increase the end-to-end delay since the destination typically has to wait longer to decode the original data packet [15]. The rate/error control advocated in [15] is based on an optimization problem. This approach lacks taking delay into consideration. In [16] an adaptive error control scheme for real-time audio over the Internet is developed. In this work the FEC scheme is selected according to its impact on the end-to-end delay using an utility function for assessing the perceived audio quality that consider the effect of the end-to-end delay. These error control schemes were designed to resolve audio packet loss over the wired IP networks; the packet loss process is different in wireless environments where loss may occur due to congestion or to residual bit errors at the link layer.

3 Problem Statement

The most sensitive case of multimedia traffic is VoIP. In particular the delay is most critical in VoIP applications. It is recognized that the end-to-end delay has a great impact on the perceived quality of interactive conversations with a threshold effect around $150ms$ [5]. For intra-continental calls, the packet delay is on the order of $30ms$ and for inter-continental calls the delay can be as large as $100ms$ [21]. While reducing the effect of a small jitter can be realized by a playout buffer introduced at the receiver, the avoidance of a high jitter/delay is much more complex. Especially retransmissions and contention-based medium access schemes are accountable for high delays and jitters. We consider the case of VoIP calls that are transmitted over heterogeneous wired/wireless networks, we assume that the wired Internet part is error-free and congestion-free and that the bandwidth bottleneck is the last-hop wireless link (Figure 1). In this case, all the voice traffic needs to be routed through the 802.11e QAP (the "bridge" between the wired and wireless networks). Hence, the QAP becomes heavily loaded, especially when the number of active stations is fairly large. Moreover, 802.11e EDCF mode grants different priorities to specific traffic classes (i.e., latency sensitive traffic) but not specific nodes. The VoIP packets may be queued at the QAP if it cannot gain TXOPs from the competition with other nodes, and will become a bottleneck in the network and this will result in additional delays. Three different types of degradation may occur in the last-hop wireless

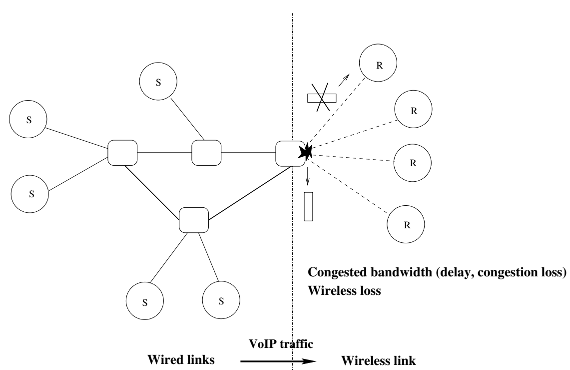

Fig. 1. Voice traffic transmitted from wired network through a last hop congested wireless link

link: packet loss due to congestion, delay due to congestion and packet loss due to bit errors at the link layer. Although a lot of voice codec can tolerate some small loss without severe degradation, most of them operate under preset schemes for data and channel code rates making them vulnerable to the varying conditions on wired and wireless IP-based hops [17]. Some kind of adaptation is therefore needed to dynamically adapt the codec bit rate to the changing wireless network conditions so as to preserve acceptable levels of reliability and quality.

4 Proposed Architecture and Rate Control Mechanism

4.1 VoIP Gateway Interworking Wired and Wireless Network

The proposed architecture (Figure 2) uses a VoIP gateway located at the edge of the wired network and the wireless link, to transcode voice flows before their transmissions over the wireless channel. The gateway is associated with an 802.11e QAP. A QAP is required to support VoIP calls between wired and wireless networks. In such a situation, the functionality of HC (Hybrid Coordinator) is performed at the QAP. The QAP may gain high priority to access the channel by piggybacking data packets on the QoS-Poll packets or the QoS Ack packets, and thus speeds up dispatching packets from wired networks. The instantaneous bit rate is determined by an adaptation algorithm (described in section 4.2) based on the estimation of wireless channel congestion status. Congestion control information can be obtained through RTCP reports sent back to sources via the HC.

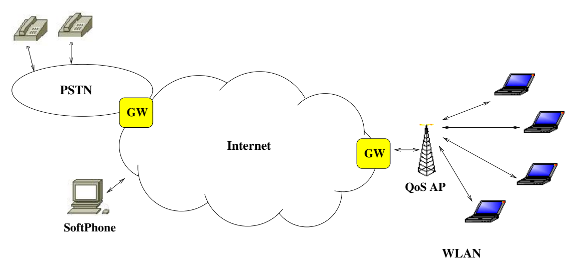

Fig. 2. VoIP gateway at the edge of wired and wireless network

4.2 Vegas-Like Audio Rate Control Mechanism

The proposed rate control mechanism is based on a TCP Vegas-like congestion avoidance technique for the rate and loss control of VoIP flows over the WLAN. The rate of the audio codec used for transcoding the voice flow at the VoIP gateway is varied according to the RTT measured between the QAP and wireless stations. In case of packet loss, a delay-based loss predictor is used to determine the type of loss and apply the appropriate strategy depending on whether packet losses are due to network congestion (transcode the voice flow using a lower audio codec bit rate) or wireless link errors (increase robustness by adding FEC). The source and channel adaptation algorithm residing at the gateway will converge to the available bandwidth in the WLAN while attempting to optimize overall call quality of several simultaneous voice communications. The input of the algorithm is the estimation of current WLAN congestion state given by delay and loss parameters. The basic idea of our Vegas-like audio rate control algorithm is to adapt the audio codec rate by varying audio packet size to avoid congestion, and this unlike TFRC mechanism that uses fixed size packets and varies the sending rate in packets/sec. The VoIP gateway keeps track of the *BaseRTT* defined as the minimum of all RTTs measured on the WLAN using RTCP receiver reports. When a receiver report related to the voice flow i is received at the gateway, the *Expected Audio Data* and the *Actual Audio Data* are calculated as:

$$ExpectedAudioData = R_i \times BaseRTT \tag{2}$$

$$ActualAudioData = R_i \times RTT_i \tag{3}$$

where R_i is the audio codec bit rate used for voice flow i. RTT_i is the round-trip time between the gateway and the wireless station i estimated when the receiver report i is transmitted through the gateway. *Actual Audio Data* represents the amount of audio transmitted during RTT_i using codec rate R_i. The difference *Diff* is calculated as:

$$\begin{aligned} Diff &= ActualAudioData - ExpectedAudioData \\ &= R_i(RTT_i - BaseRTT) \end{aligned}$$

Diff is an estimation of the extra audio data the voice flow i has in transit, i.e. data that would not have been sent if the audio codec used for this voice flow exactly matched the available wireless channel bandwidth. The algorithm then

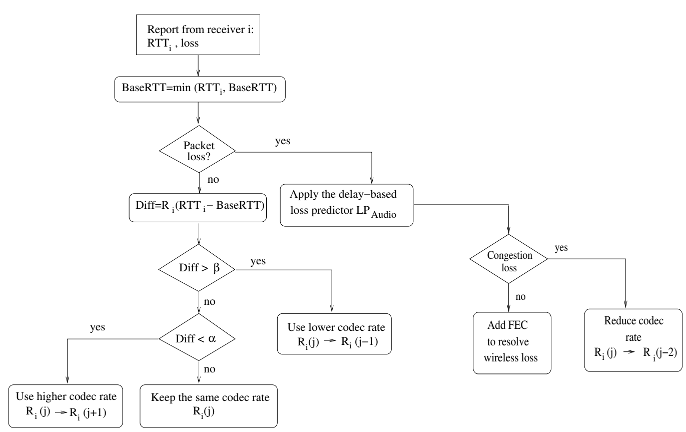

Fig. 3. Flow-chart for the Vegas-like audio rate control

compares the value of *Diff* to the α and β thresholds, these two thresholds are defined in terms of bytes. The farther away the actual audio data gets from the expected value, the more congestion there is in the WLAN, which implies that the sending codec rate should be reduced. This decrease is triggered by the β threshold. The α threshold triggers the increase of the audio codec rate in case the voice flow is not utilizing the available bandwidth. The goal is then to keep between α and β extra bytes transmitted over the wireless channel (Figure 3). In order to differentiate between congestion and wireless losses, we define the following delay-based loss predictor:

$$LP_{Audio} = \frac{Diff}{BaseRTT} = R_i(\frac{RTT_i}{BaseRTT} - 1) \quad (4)$$

LP_{Audio} would predict that next packet loss will be due to congestion when the Vegas-like audio rate control algorithm suggests that audio codec rate be decreased. If a loss occurs when the algorithm is recommending increasing codec rate, it may be reasonable to assume that the loss is due to transmission errors on the wireless channel and thus FEC information will be added in order to resolve this loss (Figure 3).

5 Simulation Experiments and Discussion

We provide $NS-2$ simulation results obtained from downlink VoIP flows transmitted on a WLAN with CBR background traffic using the EDCF/HCF mode of operation. We consider a high-rate IEEE 802.11a WLAN with physical data rate of 36Mbps and an adaptive system in which sources can switch between five bit-rates, corresponding to widely used telephone speech coding standards (Table 1). The G.726 [7] codec makes a conversion of a 64 kbit/s pulse code

Table 1. Codec bit rate and packet size

Codec	Bit Rate (Kbit/s)	Payload Size (bytes)	Total Packet Size (with IP/UDP/RTP headers)
G.711	64	160	200
G.726	40	100	140
G.726	32	80	120
G.726	24	60	100
G.726	16	40	80

modulation (PCM) channel to and from a 40, 32, 24 or 16 kbit/s channel. The conversion is applied to the PCM bit stream using an ADPCM (Adaptive Differential Pulse Code Modulation) transcoding technique. The relationship between the voice frequency signals and the PCM encoding/decoding laws is fully specified in Recommendation G.711 [6]. A variable bit-rate system operating at such bit-rates can be viewed as a system that always delivers “toll quality,” but with different levels of complexity, delay and robustness. Codec rates and packet sizes used to simulate our mechanism are shown in Table 1. For TFRC flows, we use packets of 200$bytes$ and for background flows we set packet size to 1500$bytes$. The goal of the simulations is not a complete analysis of the considered system but, rather, an indication that interesting performance evaluation indices can be derived through the proposed approach. We consider only the adaptive codec rate part of the Vegas-like control algorithm. In future work we will consider adding FEC information based on the Loss Predictor defined in (4). Simulations are carried out for the duration of 20 seconds and the presented results are averaged over 5 simulations. We set the EDCF VoIP flow priority to 6 and background flows priority to 1. The number of stations is increased from 2 to 24 (including the QAP). A VoIP and a background flow are transmitted from the QAP to each QSTA. In order to ensure more accurate responsiveness to the channel load, we use variable values for α and β parameters of our Vegas-like audio rate control algorithm that depend on an estimation of the number of audio packets transmitted by the voice flow i during the RTT_i :

$$\alpha = 30 \times \frac{RTT_i(ms)}{20ms}(Bytes)$$
$$\beta = 50 \times \frac{RTT_i(ms)}{20ms}(Bytes)$$

Figure 4 shows the average delay of VoIP traffic over the WLAN. Adaptive audio rate control presents good performance, as it keeps average delay below 4ms in both situations of small and large number of VoIP flows. With TFRC, VoIP

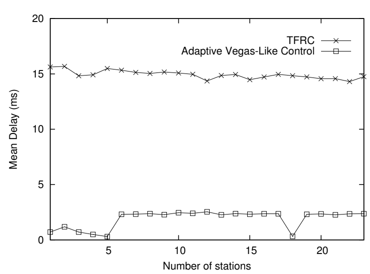

Fig. 4. Mean VoIP delay vs. the number of QSTAs

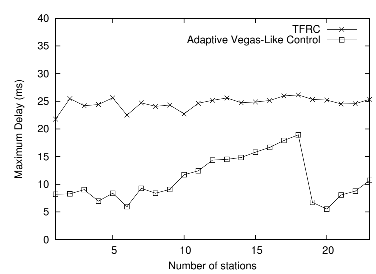

Fig. 5. Maximum VoIP delay vs. the number of QSTAs

average delay rises above 15ms. The confirmation of these results is provided by the maximum voice packet delay depicted in Figure 5. Adaptive audio rate control is able to keep the maximum delay below 10ms when the number of VoIP flows is less than 10, and below 20ms in the case of more than 10 VoIP flows. For TFRC the maximum delay is about 25ms. Reducing the audio packet delay by the value of about 10ms on the WLAN is important in order to cope with the before mentioned audio QoS requirements (one way delay is restricted to at most

150ms) and since we have to consider the delay caused in the wide area network that must be traversed by an audio packet on its way to the destination in the WLAN. The adaptability of our control mechanism to the WLAN conditions ensures a reduced packet voice delay and this improves perceived voice quality.
Figure 6 depicts the mean bandwidth of VoIP flows as the number of stations is increased. When the number of flows is below 8, the mean bandwidth of our adaptive VoIP mechanism is higher than that of TFRC. Our mechanism is less

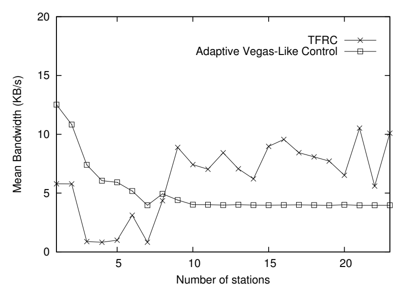

Fig. 6. Mean bandwidth of VoIP flows vs. the number of QSTAs

conservative than TFRC when the number of flows is reduced, and this avoids the under-utilization of network bandwidth and also avoids the voice transmission quality degradation in response to congestion. For more than 6 flows, the mean bandwidth used by our mechanism is steady (4 KB/s) and smoother than TFRC mean bandwidth. Maintaining low sending rate variation and avoiding abrupt rate changes will reduce the delay jitter and this will ameliorate the perceived voice quality. Besides, reducing the sending rate in case of high load network conditions will increase the WLAN capacity.
In order to evaluate the fairness between VoIP flows, we compute for each scheme the fairness index defined as:

$$FairnessIndex = \frac{(\sum_{i=1}^{n} T_i)^2}{n \times \sum_{i=1}^{n} (T_i)^2} \quad (5)$$

where n is the number of flows using the same control scheme, and T_i is the throughput of flow i. The fairness index is equal to 1 if all T_i are equal (highest degree of fairness between flows). Figure 7 shows that our adaptive audio rate control achieves considerably better fairness than TFRC. With our mechanism, fairness index is kept almost at 1 independently of the number of VoIP flows, however with TFRC, this index goes below 0.4 for a small number of VoIP flows (6 flows) and it is improved when the number of flows is increased (0.62 for 20 flows). This can be explained by the fact that our adaptive control uses delay information to avoid congestion and adapts the audio rate of each flow, however

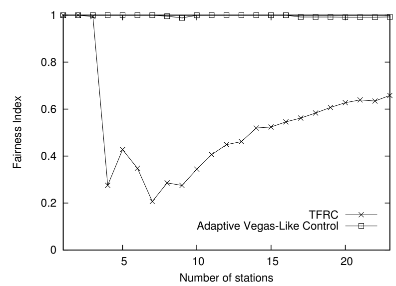

Fig. 7. Improved VoIP flow fairness with the adaptive Vegas-like codec rate control

TFRC uses loss information for rate adaptation and this information is not so accurate for detecting congestion in WLANs.

6 Conclusion

In this paper we propose a novel adaptive architecture for the transport of VoIP traffic over heterogeneous wired/wireless Internet environments. This architecture supports adaptive VoIP coding on WLAN using a VoIP gateway located at the edge of the wired Internet and the wireless network. The adaptive coding mechanism is illustrated considering a specific control mechanism applied to variable bit-rate system operating at five VoIP coding bit rates (64, 40, 32, 24 and 16 Kbit/s). Simulation results show that our adaptive architecture responds constructively to network congestion and improves QoS support for VoIP in IEEE 802.11e networks. Using the 802.11e EDCF/HCF scheme, we reduce the transmission delay of VoIP traffic compared to current TFRC algorithm. Obtained results show that our adaptive rate control mechanism is fairer than TFRC especially when the number of VoIP flows is increased. The system capacity is also increased, since the sending rate is reduced in case of high-load network conditions.

Acknowledgments

This work has been supported by the french ministry of industry in the context of the national RNRT project VTHD++.

References

1. IEEE Std 802.11-1999, Part 11: *Wireless LAN MAC and Physical Layer specifications.* Reference number ISO/IEC 8802-11:1999(E).
2. M. Veeraraghavan, N. Cocker, and T. Moors. *Support of Voice Services in IEEE 802.11 Wireless LANs.* In Proceedings of INFOCOM 2001, Alaska, April 2001.

3. D.-Y. Chen, S. Garg, M. Kappes, and K. S. Trivedi. *Supporting VBR VoIP traffic with IEEE 802.11 WLAN in PCF mode.* In Proceedings of OPNETWork 2002, Washington D. C., August 2002.
4. IEEE 802.11e/D4.0. *Wireless MAC and Physical Layer specifications: MAC Enhancements for QoS.* November 2002.
5. ITU-T. *One-Way Transmission Time.* Recommendation G.114, February 1996.
6. ITU-T. *Pulse code modulation (PCM) of voice frequencies.* Recommendation G.711, November 1988.
7. ITU-T. *40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM).* Recommendation G.726, December 1990.
8. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. *RTP: A Transport Protocol for Real-Time Applications.* Request for Comments 1889, January 1996.
9. S. Floyd, and K. Fall. *Promoting the Use of End-to-End Congestion Control in the Internet.* IEEE/ACM Transactions on Networking, August 1999.
10. J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. *Modeling TCP Throughput: A Simple Model and its Empirical Validation.* In Proceedings of ACM SIGCOMM'98, Vancouver, Canada, August/September 1998.
11. S. Floyd, M. Handley, J. Padhye, and J. Widmer. *Equation-Based Congestion Control for Unicast Applications.* In Proceedings of ACM SIGCOMM'2000, pp. 43-56, Stockholm, Sweden, August/September 2000.
12. M. Handley, S. Floyd, J. Padhye, and J. Widmer. *TCP Friendly Rate Control (TFRC): Protocol Specification.* Request for Comments 3448, January 2003.
13. S. Cen, P.C. Cosman, and G.M. Voelker. *End-to-end Differentiation of Congestion and Wireless Losses.* In Proceedings of Multimedia Computing and Networking (MMCN) conf. 2002 pp. 1-15, San Jose, CA, January 2002.
14. M. Chen and A. Zakhor. *Rate Control for Streaming Video over Wireless.* In Proceedings of INFOCOM'04, Hong Kong, March 2004.
15. J-C. Bolot, S. Fosse-Parisis, and D. Towsley. *Adaptive FEC-based Error Control for Internet Telephony.* In Proceedings of INFOCOM'99, New York, March 1999.
16. C. Boutremans and J. Y. Le Boudec. *Adaptive Delay Aware Error Control for Internet Telephony.* In Proceedings of 2^{nd} IP-Telephony workshop, Columbia University, New York, April 2001.
17. J. Matta, C. Ppin, K. Lashkari, and R. Jain. *A Source and Channel Rate Adaptation Algorithm for AMR in VoIP Using the Emodel.* In Proceedings of NOSSDAV'03, San-Francisco, California, June 2003.
18. R. Rejaie, M. Handley, and D. Estrin. *RAP: An End-to-end Rate-based Congestion Control Mechanism for Realtime Streams in the Internet.* In Proceedings of IEEE INFOCOM'99, New York, March 1999.
19. D. Sisalem and H. Schulzrinne. *The Loss-Delay based Adjustment Algorithm: a TCP-friendly Adaptation Scheme.* In Proceedings of NOSSDAV'98, Cambridge, UK, July 1998.
20. I. Rhee, V. Ozdemir, and Y. Fi. *TEAR: TCP Emulation at Receivers – Flow Control for Multimedia Streaming.* NCSU Technical Report, April 2000.
21. M. Karam and F. Tobagi. *Analysis of the Delay and Jitter of Voice Traffic over the Internet.* In Proceedings of IEEE INFOCOM'01, Anchorage, AL, April 2001.