VoIP Codec Adaptation Algorithm in Multirate 802.11 WLANs: Distributed vs. Centralized Performance Comparison

Anna Sfairopoulou, Carlos Macián, and Boris Bellalta

Network Technologies and Strategies (NeTS) Research Group
Universitat Pompeu Fabra
Passeig de Circumval.lació, 8, 08003 Barcelona, Spain
{anna.sfairopoulou, carlos.macian, boris.bellalta}@upf.edu

Abstract. Multirate 802.11 environments are quite problematic for VoIP traffic, with the rate changes of some of the flows affecting the transmission of all others. Building upon our previous results, we propose an algorithm which, based on the combined feedback from Real-Time Control Protocol (RTCP) packets and the MAC layer, can dynamically adapt the codecs of ongoing VoIP calls to adjust them to the multirate scenario. A comparison of both the centralized and distributed versions of the algorithm is provided for a wired-wireless scenario, showing an important capacity and quality increase over the standard case.

1 Introduction

Wireless Voice over IP is becoming increasingly popular as an economic alternative to traditional fixed telephony or even to cellular telephony. Nevertheless, in spite of the equal success and evolution of both VoIP and wireless networks as two separate areas, there are still problems when trying to provide voice services over 802.11 networks. The strict QoS requirements of voice transmission are often unaccomplished due to the specific nature of wireless environments, like unfairness between uplink and downlink streams [6], VoIP degradation in presence of TCP flows [1] or variable capacity due to multi-rate transmissions [2].

Multi-rate transmission is one of the key features of the IEEE 802.11 [3] PHY/MAC specifications which allow each mobile station (STA) to select its physical layer parameters (modulation and channel coding) to optimize the bit transmission over the noise/fading-prone channel. In this environment, sporadic rate changes occur to one or more STA, produced by such effects as increases in distance between the two wireless end-points, presence of walls when entering a room, atmospheric issues (rain, etc.). However, when a STA reduces its rate, the saturation point for the shared channel is reduced, affecting all active calls independently of whether they observe good or bad channel conditions [2]. As a consequence, a rate change produces a general degradation on the performance of the system, with active calls being dropped and reducing the free space for new ones, which result in higher blocking probabilities, provoking an overall

A. Pras and M. van Sinderen (Eds.): EUNICE 2007, LNCS 4606, pp. 52–61, 2007.

suboptimal allocation of the network resources. This effect can be observed in Figure 1, where the feasible distribution of simultaneous VoIP calls in a cell is depicted: considering the use of the $G.711$ voice codec for two types of VoIP calls, the *fast* and the *slow* ones, which use a data rate of 11 Mbps and 1 Mbps respectively, the maximum number of active calls drops as fast calls change to slow calls. For example, with 9 VoIP calls active, if just 1 call changes to the lower rate, the new state becomes not feasible, provoking that all active calls perceive a QoS degradation, as they can not obtain their required bandwidth. Figure 1 has been obtained using the DCF (Distributed Coordination Function) analytical model presented in [1] and adapted for the multi-rate case.

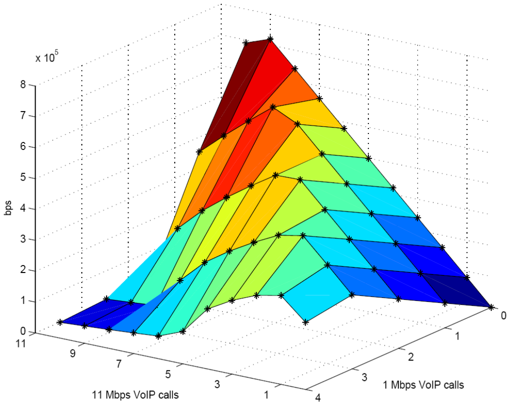


Fig. 1. Bandwidth obtained by the active calls in the presence of fast and slow calls

To the best of our knowledge, there has been few previous work on the problems that multirate wireless LANs introduce to VoIP. Some similar solutions proposing codec adaptation are the ones mentioned here. The authors in [18] focus on the Access Point (AP) acting as a bottleneck problem and propose a centralized element which performs transcoding of incoming calls to the wireless network. The work presented in [17] measures cell congestion and changes the codecs of *all* active calls accordingly, while performing admission control for new calls. Our solution is much lighter in processing effort than transcoding, can be implemented equally in a distributed way since it does not depend on a central server and only *some* of the calls need to adapt their codec. In [9] a similar mechanism to ours is proposed, but there is no consideration of cross-layer information, neither of the specific problem caused by the multirate environment. Another group of related works are the ones focusing on multirate codecs, like AMR (Adaptive MultiRate) in [8] and [14]. AMR was designed mostly for GSM/UMTS networks, where bad channel conditions refer to the ones perceived individually on each STA's channel. In the working scenario studied here, the behavior of other nodes (which change from *fast* to *slow* rates) is what impacts over the system's performance, without any channel change occurring on the fast nodes. Finally, many of the current works focus on changes of the MAC specification, like the one in [7], which proposes a new media access scheme in order to improve the capacity of the wireless cell. However, these solutions require changes on the 802.11 MAC standard while the solution proposed here can be implemented easily without any modifications on the MAC or any other protocol.

This paper proposes a Codec Adaptation Algorithm (CAA), which based on cross-layer information from RTCP and MAC layer can monitor the quality of the active calls and change their codec to adapt them to the multirate environment. Continuing the work already presented in [15][16], the focus here is on a performance evaluation of the different implementations of the CAA; centralized installed on the AP and distributed placed on each STA. Both implementations share the same basic core functions, with small changes between the two that will be explained later.

2 Codec Adaptation Algorithm

The Codec Adaptation algorithm is composed of three main phases: the *monitoring phase*, the *adaptation phase*, and the *recovery phase*, presented below very briefly due to space constraints. A more detailed explanation of the different phases can be found in [16].

Monitoring: Starting with the monitoring phase, the two procedures, MAC and RTCP monitoring, work simultaneously, each one focusing on the problem from a different angle; while MAC monitoring provides a proactive measurement, proposing to take action *before* any quality alarm arrives, RTCP monitoring responds to the alarm signals *after* they happen, in a reactive way. During the MAC monitoring the algorithm reacts on MAC alarms over rate changes. Since just one rate drop can provoke a big impact on all the calls of the cell, the moment that such event is detected the node drops immediately to a one-step lower codec in the codec ranking (codecs are ordered based on their bitrate as seen in chapter 4). This way we obtain a faster reaction to situations that are most probably going to provoke a system degradation and also a more fair solution, since the first to suffer the consequences of the rate change is the node that provoked it. The same procedure works also when there is a rate increase; the node that detects its rate changing to a higher one, changes proactively to a one-step higher rate codec, to fully use its new rate and as a result, obtain better quality on voice transmission.

If the codec change invoked by the MAC layer was not enough to re-stabilize the situation on the cell or if some other cause (like wireless channel errors) is causing additional problems, then this will be detected in the RTCP filtering. The information obtained from the RTCP packets includes critical parameters for voice traffic, such as the number of lost packets and end-to-end delay. These data are used, in order to calculate the R-factor of each flow (a real-time QoS metric proposed by International Telecommunications Union [5]). The equivalence of this factor to the better known Mean Opinion Score (MOS) can define a first QoS decision threshold; if R-factor value falls below 70 (equivalent to MOS 3.6) the quality of the voice flow is not satisfactory and the algorithm triggers the adaptation phase for a new codec selection.

Adaptation: Passing to the adaptation phase, a random timeout is set, during which the node sends a minimum of N fast RTCP packets (with N set to 3 in the simulations) in shorter intervals than the regular ones, in order to collect additional information and make sure that the alarm situation still continues. This timer is also necessary so that not all nodes will react simultaneously, even if all of them perceive the alarm signals at the same instant. This way, the change of codec for more calls than necessary is avoided and the system has some time to recover after every codec change. The interval between the transmission of the "fast RTCP packets" can be set to be lower than the standard 5 seconds interval of RTCP transmission, as defined in the proposal presented in [13] (in our simulation is set to 1 sec). This way the total process delay can be minimized to a few seconds only. We have calculated an estimation of the total overload that these packets would introduce at the network and found that the overhead provoked by the control traffic is very small (around 1.2%) compared with the data traffic of a VoIP call using $G.711$ codec.

When the timer expires the algorithm calculates the average of the R-factor from the information collected during this time and from there the MOS, as also the average of packet loss and delay. CAA compares the average values as well as the current values of the parameters after the timeout against a set of thresholds, chosen using the common values of permitted QoS parameters for an acceptable voice transmission ($delay < 150ms$, packet loss< 3%, $R > 70$). From the result of this comparison and the codec used until now in the transmission, the node can choose a new codec, using the following procedure:

a) if the average value of the parameter is out of the threshold then check its current value; if the current value is also out of threshold propose a codec of α steps lower in the codec ranking, otherwise propose a β steps lower codec.

b) if both the average and the current value of the parameter are above the thresholds then there is no need to change the codec.

This comparison is performed for each one of the three parameters (delay, loss, R) used in the evaluation and from each comparison a codec is proposed. The codec finally chosen is the average of the three proposals. Note that $\alpha > \beta$, with $\alpha = 2$ and $\beta = 1$ in the simulations, and that the codecs are ordered based on their bit rates, as explained in the results section.

Recovery: After the new codec selection, the node is responsible for issuing and sending a re-Invite SIP message to the other end, so as to renegotiate the new call characteristics. If the other end accepts the new codec the call continues normally, otherwise the call is dropped. If the codec proposed as the most appropriate is lower than the lowest codec that a node can support, the call can continue as it is during some stand-by time, without any codec change. If during this time some other node changes codec and the problem is solved then the call can continue successfully, otherwise the call will be dropped. Especially in the centralized version, the AP can choose to change the codec of another call since the present call cannot change any further. Although this way a call drop can be avoided, this extra waiting time would introduce more delay in the recovery process. This part of the algorithm has not yet been tested thoroughly and is part of the ongoing work.

3 Centralized vs. Distributed Architecture

The CAA presented above can be implemented both in a distributed and in a centralized mode with small changes between the two implementations, but with visible difference in performance and results. In the distributed scenario, the algorithm is located on each node and each node monitors its own state. When a rate change is noticed on the MAC layer or based on the RTCP information that is arriving to it, the node is the one to determine whether or not to change codec. This is a more local approach and the node can only decide on the calls depending on it, which may not be the globally optimal solution. As we have proved in our previous work [15], there is no need for all calls to change codec at the same time, and changing slow-rate calls gives better results than changing fast-rate calls. This is due to slow-calls being the ones actually causing the problem, as seen in the problem statement. In the distributed implementation the algorithm cannot give priority to the slow calls, other than the one given by the proactive change, as the call that detects first the QoS drop will be the first to react. Additionally, since the control of the waiting timer is not centralized, more nodes can coincide and change simultaneously codec. Nevertheless, the distributed approach is easy to implement and it distributes the processing load of the algorithm.

In the centralized case, the AP is in charge of monitoring all calls, the transmission rate of each flow and the codec used by each client. When a call passes from fast to slow then the AP, apart from changing the codec of it in a proactive way, can also determine which and how many other calls must change codec so as to reach network stability again, based on the RTCP information exchange between the clients. Intercepting the RTCP packets in their way from one end to the other, it calculates the instantaneous MOS value for all calls. When these values fall below the thresholds, the AP chooses the calls with the worst performance and decides which calls to change and to which codec, giving more weight and priority in changing the slow calls first. Between each codec change, the AP waits during a random time, which permits that less number of calls will have to change. The drawback of this implementation is the amount of processing work for the AP. It must intercept all RTCP packets between the two ends of the voice flows and filter them to obtain the information it needs. When there is a codec change decision, it must inform the node that there is the need to change codec and therefore suggest to issue a SIP re-Invite message to renegotiate the codec with the other end. This can be more complicated than in the distributed version, but on the other hand there is a better overall monitoring of the whole network and there are more possibilities of achieving an optimal codec combination among the nodes. Simulation results show that there is an improvement in the performance of the algorithm when used in its centralized version; We have less calls changing codec, the packet loss percentage is almost zero and the overall MOS achieved is higher than in the distributed implementation. These results will be reviewed in the following section.

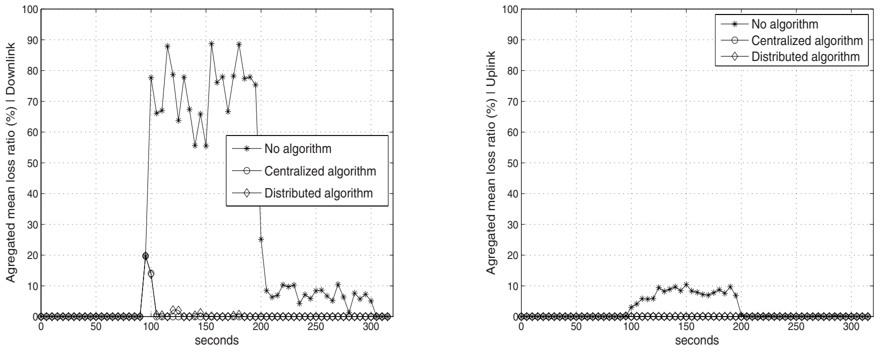
4 Simulation Results

The scenario considered is a hot-spot multirate scenario, where the network is composed by one 802.11e [4] basic service set with 9 wireless nodes and one AP connected to the wired network. A total number of 9 calls are active during the simulations, with all of them established between one wired and one wireless client, while the AP is also acting as a Proxy Server. All nodes start by using 11 Mbps data rate (fast-rate calls) and at the instant $t = 95$ of the simulation two nodes change to 1 Mbps data rate (slow-rate calls). The calls are considered to start with the G.711 codec and have the same duration. Monitoring frequency for the MAC monitor is set to 5 seconds, equal to the normal RTCP monitoring frequency. The fast RTCP transmission interval is set at 1 second, in order to minimize the reaction time of the algorithm. All users are considered to support all codecs and there is no other traffic or other interferences in the wireless network. Each STA has a queue length of $K = 50$ packets. The codecs used during simulations are the G.711, G.726, G.729 and G.723.1, ordered by their bitrate.

The network simulator $NS-2$ [10] was used for the simulations, with the addition of the SIP module obtained from [11] in order to include the basic SIP agents. It was further adapted for the specific $NS-2$ scenario with the goal of controlling the codec of each call while the call is in progress. The basic 802.11e MAC module was used, obtained from [12] with the default parameters set and the experiments were performed for different channel rates, from 1 to 11 Mbps.

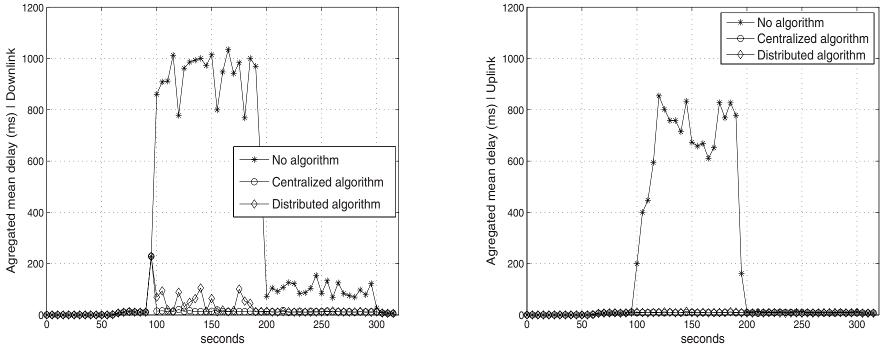
4.1 Analysis

In order to understand the efficiency of the codec adaptation algorithm it is important to see what exactly happens to the network when no algorithm is present. When just two nodes start transmitting on a lower rate changing from 11 Mbps to 1 Mbps (at instant $t = 95$ on Figure 2), the packet loss percentage gets quickly very high, with values reaching almost 90%, which can be translated to call drop since almost all packets are lost. Moreover, the packet delay increases to very high values reaching 1 sec, as the queue length becomes saturated (Figure 3). The congestion of the system, both in terms of loss and delay, is much more obvious in the AP, since it aggregates the traffic of *all* calls, this is why we observe the difference on the results between uplink and downlink. The resulted saturation can be also observed from the very low throughput obtained in Figure 4.(b) and the low quality perceived by the user in the MOS calculation (Figure 4.(a)). The observed MOS, as calculated in real-time using the E-model, drops to values as low as 1, meaning communication breakdown according to the MOS standard definition. The situation is corrected only when one of the two nodes



(a) (b)

Fig. 2. Average aggregated packet loss percentage (a) Downlink (b) Uplink

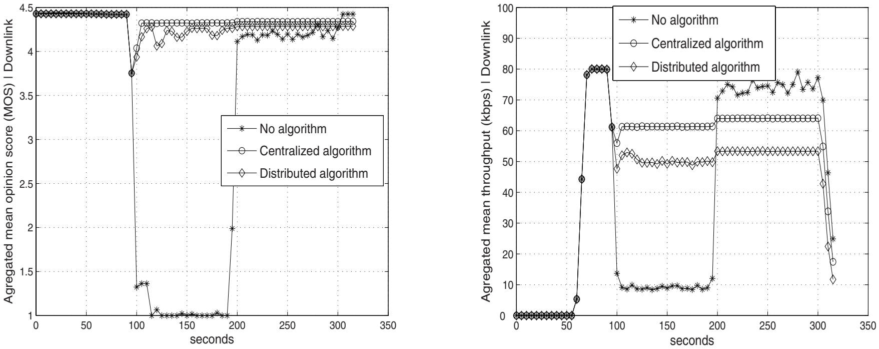


(a) (b)

Fig. 3. Average aggregated Delay (a) Downlink (b) Uplink

that previously dropped to a lower rate change again to a higher rate (11 Mbps) at simulation instant $t = 200sec$. After this point, we observe a decrease on delay and packet loss, although they remain higher than the desired for a correct VoIP transmission, with delay above 100ms and packet loss percentage of 10% in the downlink.

The best solution is to lower the congestion level of the AP by reducing the codec of some of the calls. In the distributed implementation of the algorithm, almost instantly as the rate changes happen at $t = 95sec$ of the simulation, the nodes perceive the alarm situation and after a random waiting time they adapt to the network measurements by changing the codec. As we can see from the average throughput values obtained (figure 4.(b)), only some of the calls need to change codec and the transmission is re-stabilized very quickly, so while the total throughput may be lower than before the rate changes, since some calls now use codecs that require less bandwidth, the system is no longer saturated.



(a) (b)

Fig. 4. (a) Average aggregated MOS (Downlink) (b) Average aggregated throughput (Downlink)

This can be verified in the packet loss and delay figures, where there is just one peak of high loss percentage reaching 20% and high delay of around 200 ms at the moment of the transmission rate changes, which are corrected immediately.

Even more impressive are the results of the centralized implementation. At the moment the MAC monitoring receives the rate change signal, it lowers by one the codec of the affected nodes. Along with this proactive codec change and by giving priority to the codec change on the calls that have suffered a rate drop, the total number of calls that need to change codec is lower than on the distributed implementation. This is evident from the total throughput obtained (Figure 4.(b)) which is higher than in the distributed mode, that is translated in more calls transmitting with higher bit rate codec. Again, both delay and packet loss results adjust to the expected performance as in the distributed implementation and even slightly better, with a peak at the moment of rate change which is fast corrected. During the rest of the time packet loss is practically 0 and delay remains on the order of a few milliseconds. The average MOS value, indicating the user perceived quality is maintained in very high values around 4.3, as can be seen in Figure 4.(a), with only an instant drop at the moment of the rate change and until the nodes start reacting. This shows a huge gain compared to the MOS with value less than 1.5 achieved when no algorithm is present.

As can be seen from the results, both implementations of the CAA give satisfactory performance, since no calls are being dropped, there is a fast reaction and correction of the quality degradation and there are minimal packet losses with high average MOS obtained. Comparing the centralized with the distributed implementation, it becomes clear that the centralized gives better results as expected, since the AP has an overall control of the nodes and the codec they use and provides a more efficient combination of codecs. On the other hand, this means more processing effort for the AP and the results on the distributed method are quite satisfactory and give an interesting and almost equally effective alternative to the centralized version.

5 Conclusions

In this work, the problem of a multi-rate environment on the voice over IP service was analyzed and as a solution two implementations, centralized and distributed, of the codec adaptation algorithm were proposed and compared. Based on cross-layer information from MAC and RTCP packets the algorithm can detect in real-time QoS problems on the voice flows and propose a new codec that adjusts better to the current conditions. As performance simulation results show, the algorithm proposed can solve efficiently the quality degradation provoked to the calls due to the rate changes, with no call drops, minimum packet losses and an average high MOS value.

References

1. Bellalta, B., Meo, M., Oliver, M.: A BEB-based Admission Control for VoIP calls in WLAN with coexisting elastic TCP flows. In: Koucheryavy, Y., Harju, J., Iversen, V.B. (eds.) NEW2AN 2006. LNCS, vol. 4003, Springer, Heidelberg (2006)
2. Heusse, M., Rousseau, F., Berger-Sabbatel, G., Duda, A.: RTP: Performance Anomaly of 802.11b. In: Proc. of IEEE INFOCOM 2003, San Francisco, USA (2003)
3. IEEE Std 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. ANSI/IEEE Std 802.11, 1999 Edn. (Revised 2003)
4. IEEE Std 802.11e: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. Amendment: Medium Access Control QoS enhancements, IEEE Std 802.11e (2005)
5. ITU-T Recommendation G.107: The E-model, a computational model for use in transmission planning (2000)
6. Jiwoong, J., Sunghyun, C., Chong-kwon, K.: Achieving Weighted Fairness between Uplink and Downlink in IEEE 802.11 DCF-Based WLANs. In: IEEE QShine'05, Orlando, USA (August 2005)
7. Kawata, T., Shin, S., Forte, A.G., Schulzrinne, H.: Using dynamic PCF to improve the capacity for VoIP traffic in IEEE 802.11 networks. In: IEEE Wireless Communications and Networking Conference, WCNC 2005, IEEE Computer Society Press, Los Alamitos (2005)
8. Lundberg, T., De Bruin, P., Bruhn, S., Hakansson, S., Craig, S.: Adaptive thresholds for AMR codec mode selection. IEEE, VTC Spring, Stockholm, Sweden (2005)
9. Manousos, M., et al.: Voice-Quality Monitoring and Control for VoIP. IEEE Internet Computing (July-August 2005)
10. Network Simulator-2: Version 2.28, http://www.isi.edu/nsnam/ns/
11. ns-2 SIP patch: National Institute of Standards and Technology (NIST)
12. ns-2 802.11e patch: Telecommunication Networks Group, Technical University of Berlin, Germany
13. Ott, et al.: Extended RTP Profile for RTCP-based Feedback (RTP/AVPF). Internet Draft (August 2004)
14. Qiao, Z., Sun, L., Heilemman, N., Ifeachor, E.: A new method for VoIP Quality of Service control use combined adaptive sender rate and priority marking. In: IEEE ICC'04, Paris, IEEE Computer Society Press, Los Alamitos (June 2004)

15. Sfairopoulou, A., Macián, C., Bellalta, B.: QoS adaptation in SIP-based VoIP calls in multi-rate IEEE 802.11 environments. In: IEEE ISWCS '06, Valencia, Spain, IEEE Computer Society Press, Los Alamitos (September 2006)
16. Sfairopoulou, A., Macián, C., Bellalta, B.: Dynamic measurement-based codec selection for VoIP in multirate IEEE 802.11 WLANs, Technical Report TD(07)018, Cost290 (February 2007), http://www.cost290.org
17. Tamura, T., Tadashi, I.: Wireless LAN Resource Management Mechanism Guaranteeing Minimum Available Bandwidth for Real-time Communication. In: IEEE WCNC 2005, New Orleans, USA, IEEE Computer Society Press, Los Alamitos (March 2005)
18. Trad, A., Ni, Q., Afifi, H.: Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Networks. In: Roca, V., Rousseau, F. (eds.) MIPS 2004. LNCS, vol. 3311, Springer, Heidelberg (2004)