

Fast 802.11 link adaptation for real-time video streaming by cross-layer signaling

Ivaylo Haratcherev, Jacco Taal, Koen Langendoen, Reginald Lagendijk, Henk Sips
Faculty of Electrical Engineering, Mathematics, and Computer Science, Delft University of Technology
Mekelweg 4, 2628 CD Delft, The Netherlands
{I.Haratcherev, J.R.Taal, K.G.Langendoen, R.L.Lagendijk, H.J.Sips}@ewi.tudelft.nl

Abstract—The quality of real-time video streaming over wireless links is significantly reduced due to packet losses and delays caused by rapid changes of the link conditions. We resolve this problem by two solutions. In the first place, we have developed a responsive MAC adaptation method using SNR and packet loss statistics. The second solution depends on the first one, it allows the MAC to communicate information about changes of link conditions to the video codec. Together, these solutions will prevent packet losses and skipped frames, both of which have an adverse effect on the visual quality of the decoded video. In this paper we present the results of an streaming video experiment using an 802.11a link between a fixed and a mobile station. We show that with the cross-layer signaling between the MAC-layer and the video coder, an increase of 4dB visual quality is achievable.

Keywords: video streaming, rate control, MAC layer, SNR, link adaptation, cross-layer interaction

I. INTRODUCTION

With the boom of mobile communications, IEEE 802.11 seems to become a corner-stone in wireless Internet access. Also, end-users are increasingly making use of multimedia streaming applications, like voice-over-IP and video conferencing. As a consequence, there is a demand that multimedia streaming is well supported over wireless connections. Such connections usually provide far less quality and stability than their wired counterparts. The shortcomings of wireless connections are due to two reasons: First, the radio channel is a shared medium and the frequency spectrum is limited, so both bandwidth (data rate) and transmitted power (error rate) are constrained. Secondly, in a mobile environment, the radio channel conditions change frequently because of the varying distance between stations, Rayleigh fading, and interference. As a result of these inherent problems of the wireless connections, packet delay, jitter and losses are significantly higher than in wired connections. Since multimedia applications are extremely sensitive to such packet delivery problems, mobile users experience reduced performance.

To handle the effects due to changes in the channel conditions, typically so called link adaptation (LA) is being employed at the link (MAC) level. Basically, LA is a process of automatically adjusting a number of radio/MAC parameters, so that optimal quality of packet transmission is achieved. Here optimal quality means minimizing packet loss rate (PLR) and packet delay, while keeping the data throughput as high as

possible. Too often though, LA is oriented towards download-type of applications. So the emphasis is on maximizing the throughput, while minimizing the delay and PLR is a task of secondary importance. This hinders the performance in the case of multimedia streaming.

At the application layer, the video encoder can also adapt to the link quality by changing the compression degree for example, and thus modifying the data rate. This adaptation requires that the video encoder is able to sense the link quality, for example, by getting a feedback information from the decoder side. However, such a scheme is ineffective when the round-trip delays are too long. Using this approach also introduces additional overhead.

Our solution is to combine the above methods. The video encoder will adapt based on information provided by the LA entity that is at the radio level. The information consists of the current link status report, plus some additional forecast about what the link conditions are going to be in the next couple of tens of milliseconds. In [4], [3] we already have done experiments with an adaptive video coder and MAC-layer based on QoS negotiations [5], but without tight coupling of the rate-control loops.

The remainder of the paper is organized as follows. The adaptive video coder is discussed in the next section. Section III gives a description of the basics of the link adaptation, along with our hybrid rate-control solution. The inter-layer communication is discussed in Section IV. The experimental results are discussed in Section V. The conclusions are presented in Section VI.

II. ADAPTIVE VIDEO CODING

We use a video coder that is able to adapt to a changing channel. The video codec we used, is a H.263 codec that is modified to support interaction with other protocol layers such as in this case the MAC layer. Our version of the H.263 encoder supports a (video) rate-control algorithm (VRCA) that tries to achieve a certain rate, by adjusting the quantization step size. The quantization step size is the main parameter that controls the compression of the video. Although this VRCA is designed for constant bit rate (CBR) encoding, it can also be used to dynamically change the bit rate that is produced by the encoder. Since the resulting bit rate depends on the selected quantization step size but also on the statistics of the picture

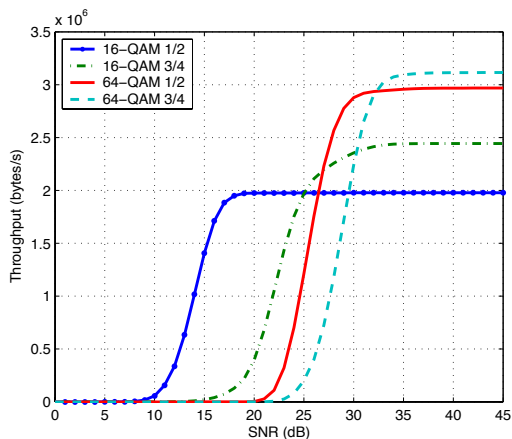


Fig. 1. Throughput vs. SNR for some 802.11a modulation schemes.

itself, we cannot — unfortunately — set the bit rate beforehand, and then expect that this bit rate will be exactly achieved by the encoder. The VRCA, is a simple feedback control loop that consists of setting an initial quantization step size, encoding part of the picture, measure the resulting intermediate bitrate, increase or decrease the step size accordingly and then continue with the next part of the picture. In total there are nine parts of a picture frame for which the quantization step size can be adjusted, which generally is enough to be able to achieve a certain preset rate. This algorithm works fine in the CBR case and with slowly varying picture statistics. In principle we can change the target bit rate for each individual frame, meaning that we have a maximum delay of 100 ms to respond to changes in the channel. In fact, we can even change the target bit rate for each part *within* a frame, so we can even faster adjust to the channel conditions. In our experiment, we will constantly adapt the target video encoding rate for the VRCA according to information from the MAC rate-control algorithm.

III. MAC LINK ADAPTATION

The IEEE 802.11 standard defines several MAC-level parameters (settings) that are available for tuning at the side of the Wireless Network Interface Card (NIC). The most important parameter available for adjustment is the transmit rate. In 802.11a, for example, it can be set to 6, 9, 12, 18, 24, 36, 48 and 54 Mb/s. Each rate corresponds to a different modulation scheme with its own trade-off between data throughput and distance between the stations. This can be seen in Figure 1 – for clarity only the last four modulation schemes are shown. It shows the performance in terms of the throughput for each modulation scheme available in IEEE 802.11a versus the signal-to-noise ratio (SNR). Note that distance is related to SNR as $SNR \sim \frac{1}{dist^\alpha}$. More complex modulation schemes like 64-QAM 3/4 offer a larger throughput, but also have increased sensitivity to channel noise, and thus provide a shorter operating range. Usually, one wants to extend the operating range as much as possible and, at the same time, to maximize the throughput. This can be done by proper (automatic) selection of the rate (modulation scheme)

that gives the maximum throughput for certain conditions, for example, by selecting 64-QAM 1/2 at an SNR of 30 dB (Figure 1).

In order to decide which rate is optimal at each specific moment, a control algorithm needs information about the current link conditions, or so-called channel state information (CSI). Since it is difficult to get CSI directly, most MAC rate-control (MRC) algorithms use some form of statistics-based feedback, for example, user-level throughput. The main disadvantage of this indirect feedback is that it is inherently slow, causing communication drop-outs when the link conditions degrade rapidly (e.g., when the user moves fast). The short-term drop-outs are normally handled by frame retransmissions. This is acceptable for download applications whose main requirement is gross data throughput. The retransmissions, however, lead to a significant increase in (average) packet delay, and the variations in the number of retransmissions cause an increase in the jitter of the packet delay. Streaming applications are very sensitive to long packet delays and high jitter, and less sensitive to the overall throughput of the link (provided of course that this throughput is larger than the minimum throughput that the application requires). Consequently, streaming applications perform poorly under standard automatic MRC. In [1], for example, it is reported that switching from automatic MRC to a manually selected fixed rate extends the maximum distance between stations up to 40% for the flawless display of a video stream over an 802.11a wireless connection.

A logical way to cope with the slow accommodation characteristics of statistics-based feedback methods is to look for methods that use faster feedback, i.e., feedback that quickly provides up-to-date information about the channel status. Such a feedback — the SNR — has been theoretically discussed in previous works, but so far, to our knowledge, it has been used in only one practical implementation [2]. We use this LA control algorithm in our 802.11 radio to enhance multimedia performance, and also to provide feedback information about the channel conditions that the video encoder requires.

IV. LAYER INTERACTION

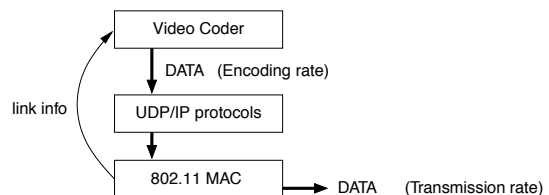


Fig. 2. Video layer, UDP/IP layer and the MAC layer. The MAC gives link information to the video layer at request of the video layer.

In our realtime video transmission scenario, the video rate control algorithm depends on information from the MAC rate control algorithm. By coupling the rate control algorithms of the MAC and the video coder in this manner we expect to efficiently use the available transmission rate to maximize the picture quality.

In the envisioned scenario, we typically encode a video sequence at ten frames per second. Each video frame is split

up in nine slices (a horizontal group of blocks). For each slice, the quantization step size can be changed by the VRCA. The MAC-layer rate control algorithm can inform the VRCA about the current transmission rate on a packet basis, such that the VRCA can adapt the target encoding rate for the current frame (Figure 2). This may have several drawbacks. 1) If the transmission rate fluctuates, this results in a picture with fluctuating quality. 2) The VRCA may not be able to follow fast changes in the transmission rate. Sudden large drops or peaks in the expected transmission rate, may then result in a video encoding rate that not matches the transmission rate.

Both observations led us to conclude that we need a longer term prediction of the transmission rate, besides the short-term or next-packet transmission rate. The validity of this prediction should then be in the order of one video frame or 100ms. On the one hand, we can use this long term prediction to more accurately initialize and adapt the quantization step size. On the other hand, this long term prediction is not always accurate enough. The inaccuracy is no problem since we can still adapt at a later stage, using the next-packet transmission rate. In stationary circumstances the long-term prediction will be quite accurate. On non-stationary circumstances, however, a large over-estimate of the transmission rate may result in an encoding rate that is too large, causing buffer overflows and large delays. Large under-estimates are not so bad, since the encoding rate is then lower than the transmission rate and all data will arrive on time, only resulting in a little lower picture quality than necessary. This means that we rather have an underestimate than an overestimate of the future rate. We therefore use a scheme as shown in Figure 3, where the MAC rate-control delivers two expected transmission rates. 1) The transmission rate for the next packet (accurate). 2) A conservative expectation of the average transmission rate over a time span of 50-100ms. The VRCA then uses these two values for fast adaptation of the video encoding process.

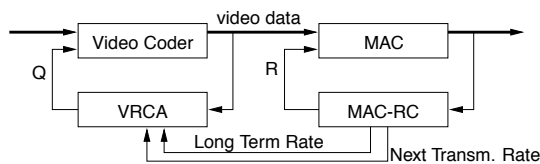


Fig. 3. Simplified overview of the coupled Rate control scheme. The rate control mechanisms of the video encoder and the MAC are coupled by the flow of information about the MAC transmission rates. For simplicity the UDP/IP layers are discarded in this picture.

On the MAC side the predictor uses history of SSIA (Signal Strength Indication of Acknowledgements) measurements to maintain a short-term (in order of few tens of milliseconds) SSIA average. Over this average future SSIA are predicted. Our experiments show that the best results are obtained by first or zero-th order predictor. This conclusion agrees with the theory as well, the latter stating that channel state change is a memoryless process.

The SSIA predictions are matched against the SSIA thresholds in the SSIA-rate lookup table in the radio rate-control algorithm [2], and so the rate predictions are determined.

Our experiments show, that for the dynamic case (the user moves intensively), the accuracy of the rate prediction for 100ms ahead is better than 80%. And the probability that the predicted rate will be within the real rate ± 1 is larger than 90%.

V. EXPERIMENTS

The experiments were carried out by streaming a video file between two laptops, both running Linux. The 802.11a cards used are based on the Atheros AR5000 chipset, and the card driver uses the advanced hybrid rate control algorithm, as described in [2]. One of the laptops had a fixed position and the other one is following a predetermined track. The track consists of three parts - "lead-in", which is reaching from the room to a specific start position in the hallway, and waiting until certain time elapses (10s). Then follows moving with the laptop three times up and down the hallway (60s). Finally we go back ("lead-out") again into the room, where the fixed laptop lies (20s).

To evaluate the performance of the coupled rate-control systems, we have evaluated three cases. Each experiment took 90 seconds. We managed to quite precisely time the experiments and to obtain a reasonable repeatability of the results. Later in this section we will compare the quality of the received videos using the Peek-signal-to-noise-ratio (PSNR), a commonly used metric in video compression.

- 1) **Coupled** : Coupled rate control. The rate control loops of the MAC-layer and the Video coder are coupled by letting the video coder poll frequently for the current and predicted rate.
- 2) **De-Coupled-MAX** : De-coupled rate control. Since the video now has no indication of the actual rate, we set the target-rate for the VRCA to the *maximum* rate as was obtained from the VRCA in the Coupled case.
- 3) **De-Coupled-AVG** : De-coupled rate control. Here we set the target-rate for the VRCA to the *average* rate as was obtained from the VRCA in the Coupled case.

In the De-Coupled-MAX case, we expect to obtain a higher quality than in the De-Coupled-AVG case. However, when the actual rate is lower than the preset target-rate, the encoder is not able to stream all data in time. The encoder will then have to skip frames¹, which will result in a lower PSNR and a lower visual quality at the receiving side. In the De-Coupled-AVG we expect an overall lower quality, since it is coded at an overall lower rate. On the other hand, we expect less skipped frames to occur.

In Figure 4 the quality (PSNR) is shown for the whole experiment (90s). In the left part (0–10s) the channel conditions are excellent, hence the high quality in all cases. The middle part is best described as having conditions changing from good to bad a few times. The right part has good conditions again.

¹The encoder has two options in this case. 1) Skip a frame such that the next frame can arrive in time, effectively lowering the rate 2) Just keep on encoding at the same rate, which will result in buffer overflows and lost packets. Both options will result in degraded visual quality. We have chosen the first option

Case	# Skipped frames	# Lost Packets	PSNR	PSNR Middle
Coupled	50	13	40.5	39.3
De-Coupled-MAX	193	19	37.8	35.4
De-Coupled-AVG	216	23	36.3	33.1

TABLE I

RESULTS FOR THE THREE CASES. THE COLUMN “PSNR” SHOWS THE AVERAGE PSNR OVER THE WHOLE SEQUENCE. THE COLUMN “PSNR MIDDLE” SHOWS THE AVERAGE PSNR DURING THE “CHANGING CONDITIONS PERIOD” BETWEEN 10 AND 70 SECONDS.

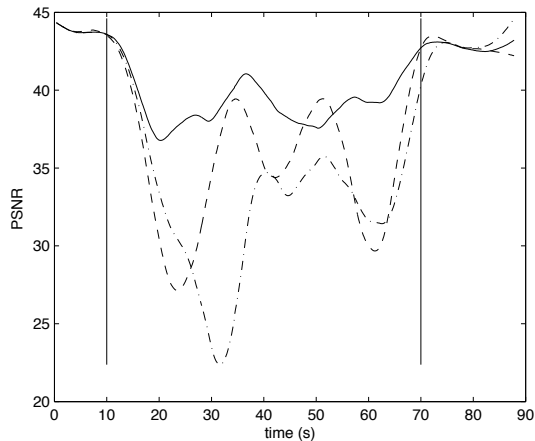


Fig. 4. PSNR: the solid line shows how the quality (PSNR) changes during the experiment for the Coupled. The dashed line shows the results for a decoupled rate control using the maximum rate (De-Coupled-MAX). The dash-dotted line shows for the decoupled rate control using the average rate (De-Coupled-AVG). We have applied an averaging sliding window to the data for better visibility.

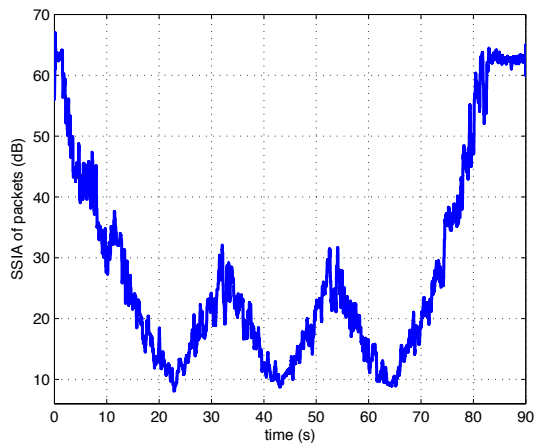


Fig. 5. Measured SSIA during experiments.

In Figure 5 the received-signal strength is shown for the same time span. The pattern clearly corresponds to the followed track. Note that the strength already drops quite strongly in the first (0-10s) part. However, down to 30 – 35dB the highest possible rate (at the radio) provides excellent performance.

As we can see in Figure 4, the Coupled case has a higher PSNR in almost all cases. Table I summarizes the number of skipped frames (by the encoder), the number of lost packets, the average PSNR and the average PSNR in the period between 10–70s. Although the number of lost packets are not the same in the three cases, the differences do not justify the differences in PSNR. In the decoupled cases, the total number of skipped frames is much higher, as we expected. Looking at the average PSNR in the 10–70s period, we conclude that the quality can be dramatically improved by informing the video codec of the actual present rate and a prediction for the near future.

VI. CONCLUSIONS

In this paper we have presented the results of a streaming video experiment over a wireless 802.11a connection between two laptops, one of which was moving. The changing channel conditions make that the video cannot always be transmitted at the full rate, without packet losses. To handle the changing conditions and packet losses we implemented two solutions. 1) responsive MAC-adaptation using (radio-)SNR and packet loss statistics. 2) cross-layer signalling of changing channel conditions between the MAC-layer and the video coder. The first solution tries to prevent packet-losses and transmit buffers running full, by lowering the transmission rate when high packet-losses are expected. The second informs the video codec of these changes such that the transmission rate is matched with the generated video data rate. As a result, our video streaming experiments suffered from a very small number of packet losses and also a small number of skipped frames. Both effects resulted in a high visual quality as opposed to the case where there is no cross-layer signalling. In changing channel conditions, an increase of 4dB PSNR or more is a realistic figure.

REFERENCES

- [1] I. Haratcherev, K. Langendoen, I. Lagendijk, and H. Sips. Application-Directed Automatic 802.11 Rate Control: Design Rationale and Preliminary Results. In *ASCI 2003*, pages 56–60, Heijen, The Netherlands, June 2003.
- [2] I. Haratcherev, K. Langendoen, I. Lagendijk, and H. Sips. Hybrid Rate Control for IEEE 802.11. In *ACM International Workshop on Mobility Management and Wireless Access Protocols (MobiWac)*, pages 10–18, Philadelphia, PA, USA, October 2004.
- [3] J. Taal, I. Haratcherev, K. Langendoen, and R. Lagendijk. Quality of service controlled adaptive video coding over IEEE 802.11 wireless links. In *ICME, special session on Networked Video*, Baltimore, MD, June 2003.
- [4] J.R. Taal, K. Langendoen, A. van der Schaaf, H.W. van Dijk, and R.L. Lagendijk. Adaptive end-to-end optimization of mobile video streaming using QoS negotiation. In *ISCAS, special session on Multimedia over Wireless Networks*, volume I, pages 53–56, Scottsdale, AZ, May 2002.
- [5] H. van Dijk, K. Langendoen, and H. Sips. ARC: a bottom-up approach to negotiated QoS. In *3rd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2000)*, pages 128–137, Monterey, CA, December 2000.