

# A WIRELESS VIDEO TRANSMISSION CONTROL APPROACH THROUGH STOCHASTIC DYNAMIC PROGRAMMING

*Víctor Miguel, Julián Cabrera, Fernando Jaureguizar, Narciso García*

Grupo de Tratamiento de Imágenes  
ETSI Telecomunicación, Universidad Politécnica de Madrid  
Madrid, Spain  
{vmb, julian.cabrera, fjn, narciso}@gti.ssr.upm.es

## ABSTRACT

This paper presents an intelligent, rate-limited multicast video transmission optimization scheme for video distribution over 802.11 wireless networks based on packet retransmissions. We propose a problem formulation which involves the characteristics of the encoded stream together with the behaviour of the wireless channel. Using standard Stochastic Dynamic Programming techniques, optimal control policies are obtained off-line. These policies are optimal in the sense of minimizing the expected distortion at the terminal. In addition, the on-line complexity of our approach is very low since the optimization problem is solved off-line. The performance of our scheme has been evaluated in a real scenario and compared with that of a limited rate ARQ algorithm. The results for our proposed system show a higher packet recovery rate and a better protection of information with a higher priority.

**Index Terms**— 802.11, RTP, ARQ, QoS, Stochastic Dynamic Programming, wireless channel, retransmission policy.

## 1. INTRODUCTION

The widespread use of 802.11 wireless networks has spawned several multimedia applications, such as wireless video distribution in the home environment [1]. However, the operation requirements of multimedia applications do not sit well with the fluctuating behaviour of wireless channels [2]. The working conditions of these channels can vary significantly during the transmission, causing random packet losses and seriously affecting the quality of service (QoS).

Among the existing works on wireless video optimization, the use of retransmission techniques, or ARQ, is more extended in limited rate environments. However, ARQ systems require feedback information from the receiver and increase the overall wireless traffic, which might cause additional losses. Some works reduce this congestion using enriched acknowledgements [3][4]. Other works negotiate the retransmission according to the importance of the packet, assigning a limited number of retransmissions for each packet [5] or selecting its buffering time [6] according to its importance.

In this paper, we define an optimization problem with a bounded retransmission rate. The solution to the optimization problem, referred as a *retransmission policy*, determines which packets have to be retransmitted in order to minimize the distortion at the receiver.

This work has been partially supported by Alcatel-Lucent and the Spanish Administration agency CDTI under project CENT-VISION 2007-1007 and by the Ministerio de Ciencia e Innovación of the Spanish Government under project TEC2007-67764 (SmartVision).

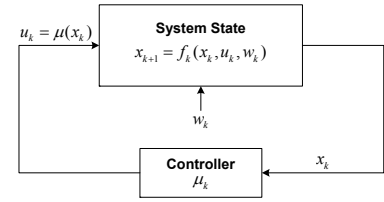


Fig. 1. Stochastic Dynamic System

The calculation of retransmission policies  $\mu_k$  is addressed from the point of view of Stochastic Dynamic Programming (SDP) [7], a mathematical tool used in previous works, among other applications, for coding rate optimization in video coder design [8] [9]. The SDP approach models the optimization problem as a discrete dynamic system, as the one described in Figure 1. At any stage  $k$ , the system is described by a state vector  $x_k$  containing variables that capture the relevant information needed to characterize system evolution.

The system is controlled by the application of actions,  $\{u_k\}$ , at each stage. At a given stage  $k$ , the system is in state  $x_k$  and an action  $u_k$  is applied. Then, the system evolves to the next stage moving to another state  $x_{k+1}$ . The evolution to the next state,  $x_{k+1} = f_k(x_k, u_k, w_k)$ , depends on the previous state,  $x_k$ , the action taken,  $u_k$ , and a random noise,  $w_k$ , that makes the behaviour of the system stochastic. In addition, the system incurs in a certain cost defined by  $g_k(x_k, u_k, w_k)$ . The selection of the appropriate action for each stage  $k$  is carried out by the control policy  $\mu_k$ . The control policy defines for each possible value of the state vector  $x_k$  which action has to be taken.

SDP algorithms compute optimal control policies provided that the optimization problem can be modeled as a discrete dynamic system. These optimal control policies minimize the cost function for  $N$  stages. If  $N$  is very large, the SDP problem can be formulated as an infinite horizon problem, where policies are stationary (independent of the stage  $k$ ), that is,  $\mu_k = \mu$ . The cost function for the average cost problem can be formulated as:

$$\lim_{N \rightarrow \infty} \frac{1}{N} E \left\{ \sum_{t=0}^{N-1} g_t(x_t, \mu(x_t), w_t) \right\}. \quad (1)$$

Our motivation for choosing SDP techniques is twofold. First of all, the random behaviour of the wireless channel makes the retransmission problem into a stochastic problem. Second, optimal policies are calculated off-line based on the models of system behaviour, thus drastically reducing the online complexity of the system.

The paper is organized as follows: in Section 2 we describe the dynamics of our system, which we use in Section 3 to formulate

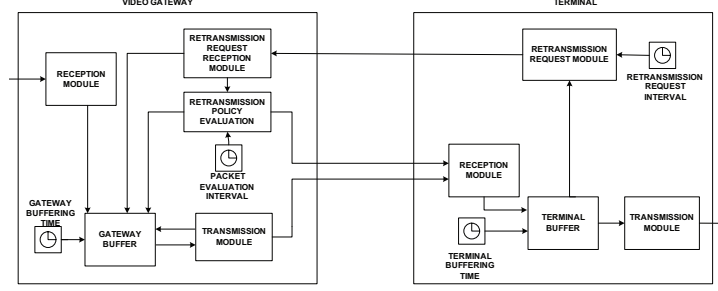


Fig. 2. System Diagram

our rate-distortion problem; in Section 4 we show and compare our results with a similarly rate-limited ARQ approach without intelligent retransmission strategies; finally, we present our conclusions and propose lines of future work in this area.

## 2. SYSTEM DYNAMICS

The formulation of the SDP problem requires the characterization of the system's behaviour in order to define both our system state  $x_k$ , our control variable  $u_k$  and their evolution. The dynamics of our proposed system, described in Figure 2, follow an ARQ scheme with a limited retransmission rate.

The Video Gateway sends RTP packets to the Terminal through the wireless link. They are also stored in the Gateway buffer until they are evaluated for retransmission. The Terminal receives these packets and stores them in its buffer, waiting for retransmission of lost packets. Using their sequence number, the Terminal also determines which packets have been lost during the transmission and sends a retransmission request containing the list of lost packets. The frequency of the retransmission request process is determined by a fixed time interval, known as the Retransmission Request Interval.

The retransmission request packet arrives at the Gateway, which marks the lost RTP packets in its buffer. The Gateway applies the retransmission mechanism periodically, with a frequency given by the Packet Evaluation Interval, for those stored RTP packets which have been stored for an interval higher than the Gateway Buffering Time interval. This parameter is implemented as a waiting time to allow the arrival of retransmission requests from the terminal.

The Gateway determines the retransmission decision for each group of  $S$  packets according to the retransmission policy (computed off-line using SDP algorithms). Once a retransmission decision has been taken for every group of  $S$  packets, the system stores the packets assigned to be resent in the retransmission buffer. The number of retransmitted packets is limited by its occupation, which has a maximum size of  $b_{max}$  packets and is emptied at a rate of  $R_{rtx}$  packets. The Gateway resends the retransmission buffer and the group of evaluated packets is erased from its buffer. Consequently, each packet has a single retransmission opportunity.

In the meantime, the Terminal has been presenting the video packets to the multimedia application after they have been stored for a certain time interval, waiting for retransmissions. This interval, or Buffer Terminal Time, plays a key role in the performance of the system. If it is too low, the retransmitted packets will arrive too late and no losses will be compensated. If it is too high, the number of recovered packets will increase, but so will the end-to-end delay.

The retransmission decisions are aimed at the minimization of the expected distortion at the Terminal, which depends on several factors. Firstly, the expected distortion is proportional to the importance of the lost information; the distortion caused by the loss of

a reference frame would be higher than the loss of a non-reference frame. This information is available in additional RTP headers supplied by Alcatel-Lucent software. Then, the behaviour of the wireless channel is also important, since it influences both the probability loss and the success of retransmission attempts. It is also necessary to take into account retransmission requests and packet probability loss, since it is possible that no request is received for a lost packet. Lastly, buffer limitations are another factor for our system model.

## 3. PROBLEM FORMULATION

Using the system dynamics, we model the behaviour of the system as a discrete dynamic system. First, we identify the variables for the system state  $x_k$  (with the aim of capturing the behaviour of the Gateway and Terminal) and the control decision  $u_k$ . Then, we use the relationships between the aforementioned variables to model the evolution of the stochastic system. Finally, we define a cost function for our system, based on the expected distortion at the receiver.

### 3.1. System Variables

Using the information provided by the description of the system, we describe system state  $x_k$  using the following variables:

- Information type,  $I_k$ : represents whether the packet contains information from an IDR frame or not.
- Channel state,  $c_k$ : shows whether last packet was successfully transmitted or not.
- Retransmission request,  $n_k$ : describes whether the evaluated packet has been requested as a lost packet or not.
- Retransmission buffer occupation,  $b_k$ : represents the number of packets in the retransmission buffer. Once it reaches the maximum capacity of the buffer, no additional packets can be added until buffer occupation decreases.
- Buffer counter,  $t_k$ : controls the retransmission rate. Its value decreases with each state transition and represents the number of stages left before emptying the retransmission buffer in  $M$  packets and restarting the counter  $t_k$ .

In the case of the control variable  $u_k$ , it represents the decision of whether to retransmit each one of the evaluated  $S$  packets or not.

### 3.2. System Evolution

The evolution of the system state is described through a set of transition probabilities calculated using the present and the previous system state,  $x_k$  and  $x_{k-1}$ , together with the previous value of the control variable,  $u_k$ . That is, we must obtain the probabilities

$$P(I_k, c_k, n_k, b_k, t_k / I_{k-1}, c_{k-1}, n_{k-1}, b_{k-1}, t_{k-1}, u_{k-1}), \quad (2)$$

where the subindex  $k$  represents the stage index. At first glance, it is obvious that  $I_k$  is independent of the other state variables and the control variable, so we can rewrite (2) as

$$P(I_k/I_{k-1})P(c_k, n_k, b_k, t_k/c_{k-1}, n_{k-1}, b_{k-1}, t_{k-1}, u_{k-1}). \quad (3)$$

The transition probability  $P(I_k/I_{k-1})$  expresses the transition between groups of  $S$  packets according to their information type. It is modeled using a first-order Markov model, which is estimated from packet traces from the transmitted video.

The variable  $c_k$ , which defines the channel state, is also independent of the rest of the variables and is modeled with a Gilbert-Elliot model. The retransmission request variable  $n_k$ , on the other hand, represents the transmission success of the  $S$  evaluated packets, which were transmitted at the previous stage  $k-1$ . For this reason,  $n_k$  depends on the previous channel state  $c_{k-1}$  and the channel model transition probabilities, as shown in

$$p(n_k/c_{k-1}) = p(n_k(0)/c_{k-1}) \prod_{i=1}^{S-1} p(n_k(i)/n_k(i-1)), \quad (4)$$

where  $n(i) = 0(1)$  represents whether packet  $i$  ( $i \in S$ ) was erroneously (correctly) received.

The evolution of the buffer occupation variable  $b_k$  and the buffer counter  $t_k$  variables is also independent of  $I_k$ ,  $c_k$  and  $n_k$ . In the case of buffer occupation  $b_k$ , its transition is only influenced by its previous state  $b_{k-1}$ , the previous control decision  $u_{k-1}$  and the buffer counter  $t_{k-1}$  according to the following rules:

- if the counter  $t_k$  equals zero,  $b_k = b_{k-1} + L(u_{k-1}) - M$ , where  $L(u_{k-1})$  represents the total number of packets added to the retransmission buffer by the decision  $u_{k-1}$  and  $M$  represents the number of packets which have been removed from the retransmission buffer.
- in the rest of cases,  $b_k = b_{k-1} + L(u_{k-1})$ .

The buffer counter variable  $t_k$  is decreased with every transition until it reaches zero, that is,  $t_k = t_{k-1} - 1$ , until it reaches zero, returning to its maximum value afterwards. The state transition probability from (2) can be rewritten conclusively as

$$P(I_k/I_{k-1})P(c_k/c_{k-1})P(n_k/c_{k-1})P(b_k, t_k/b_{k-1}, t_{k-1}, u_{k-1}). \quad (5)$$

### 3.3. Cost function

We formulate our SDP problem as an infinite horizon problem, with an average cost function  $g_k$  based on the expected distortion produced at the receiver by the decision  $u_k$ ,

$$D_k = \lim_{N \rightarrow \infty} \frac{1}{N} E \left\{ \sum_{l=0}^{N-1} D_l(x_l, u_l, w_l) \right\}, \quad (6)$$

where  $N$  represents an undetermined number of stages and  $D_l$  represents the distortion at stage  $l$ . Its value represents the sum of the individual distortion of the  $S$  packets of each evaluation stage,

$$E\{D_k(x_k, u_k, w_k)\} = \sum_{i=0}^{S-1} E\{D_i(I_k(i), n_k(i), c_{k-1}, u_k(i))\}, \quad (7)$$

where the contributed distortion for each packet  $i$  depends on the importance of its information  $I_k(i)$ , whether it has been requested

or not according to  $n_k(i)$ , the previous channel state  $c_{k-1}$  and the decision  $u_k(i)$ , which represents whether the packet was retransmitted or not. The expected distortion for each packet is calculated as

$$E\{D_i(I_k(i), n_k(i), c_{k-1}, u_k(i))\} = D_{loss}(I_k(i))P_{loss} + D_{ok}(I_k(i))(1 - P_{loss}), \quad (8)$$

where  $P_{loss}$  represents the loss probability of the packet,  $D_{loss}$  represents the distortion in the event of packet loss and  $D_{ok}$  represents the distortion for a correctly received packet.

If the decision  $u_k$  exceeds retransmission buffer limitations, the expected distortion cost  $D_i$  equals to an overflow cost  $D_{overflow}$ . In any other case, the value of  $D_{loss}$  depends on the importance of the packet; a lost packet with IDR information produces a higher distortion. For  $D_{ok}$ , we assume that a correctly received packet does not cause distortion ( $D_{ok} = 0$ ). The value of  $P_{loss}$  depends on the values of  $n_k(i)$ ,  $c_k$ ,  $c_{k-1}$  and  $u_k(i)$  as follows:

- If the packet is confirmed as lost ( $n_k(i) = 1$ ) and it is not retransmitted ( $u_k(i) = 0$ ), it is lost and  $D_i = D_{loss}(I_k(i))$ .
- If the packet is confirmed as lost ( $n_k(i) = 1$ ) and it is retransmitted ( $u_k(i) = 1$ ),  $P_{loss}$  is given by the probability of losing the retransmitted packet, thus being  $c_k = 0$ , that is,  $D_i = D_{loss}(I_k(i))p(c_k = 0/c_{k-1})$ .
- If the packet is not confirmed as lost ( $n_k(i) = 0$ ), and it is not retransmitted ( $u_k(i) = 0$ ),  $P_{loss}$  equals packet probability loss  $p_E$ , given by the stationary probabilities of the channel model. In this case,  $D_i = D_{loss}(I_k(i))p_E$ .
- If the packet is not confirmed as lost ( $n_k(i) = 0$ ), and it is retransmitted ( $u_k(i) = 1$ ),  $P_{loss}$  equals to the probability of losing the packet during both transmission ( $p_E$ ) and retransmission ( $p(c_k = 0/c_{k-1})$ ), in which case packet distortion is calculated as  $D_i = D_{loss}(I_k(i))p(c_k = 0/c_{k-1})p_E$ .

## 4. RESULTS

### 4.1. System evaluation

The resulting retransmission policies are evaluated in a real scenario implementing a multicast 802.11g network. A video flow coded in H.264 packetized in RTP/MPEG2-TS with an average throughput of 1.5 Mbps and peaks of up to 6 Mbps is received by two Terminals. We use an evaluation size of  $S = 3$  packets, in order to limit the dimensionality of the state vector space, and calculate the retransmission strategies  $u_k$  using the step-value iteration algorithm [7].

We compare our SPD-based ARQ system with a rate-limited ARQ algorithm with the exact same rate limitations. The rate-limited ARQ algorithm does not consider either information about the type of packets or the channel behaviour. If the available retransmission rate is not used up answering retransmission requests, the remaining bandwidth will be spent with random retransmissions of queued packets. Both the SDP-based ARQ and the rate-limited ARQ have a rate limit of 15% with respect to the main packet stream.

The evaluation of both systems is carried out for two situations:

- a transmission with retransmission requests, or Requests On, which represents the normal operation of an ARQ system.
- a transmission with no retransmission requests from the terminals, or Requests Off, which represents situations where no requests arrive at the Video Gateway. This could be either accidental, due to loss of the retransmission requests, or intentional, in order to avoid further congestion for multicast networks with many users.

	Rate-limited ARQ		SDP-based ARQ	
User 1	Requests Off	Requests On	Requests Off	Requests On
Overall Recovery Rate	3.11%	3.13%	9.91%	16.58%
IDR Recovery Rate	3.36%	3.75%	6.22%	10.53%
User 2	Requests Off	Requests On	Requests Off	Requests On
Overall Recovery Rate	2.98%	3.07%	7.59%	14.93%
IDR Recovery Rate	3.09%	3.67%	8.56%	11.87%

**Table 1.** Preliminary results for Buffer Terminal Time=1000 milliseconds

The performance of both systems is measured based on the overall recovery rate, defined as the percentage of recovered packets in relation to the amount of lost packets, and the IDR recovery rate, defined as the percentage of recovered packets containing IDR information in relation to the total amount of lost IDR packets.

#### 4.2. Results

Initial results, presented in Table 1 for a Terminal Buffer Time of 1000 milliseconds, were obtained for transmission experiments in our implemented 802.11g video delivery network. According to our previous experiments, this is the most sensible Terminal Buffer Time which enables the reception of retransmissions at our receiver while keeping a limited latency value. All the transmission experiments suffered a packet loss rate between 1% and 3%.

The recovery rates for both systems are higher when retransmission requests are available, as shown in the Requests On results from Table 1. However, the rate-limited ARQ system offers a lower recovery rate than our SDP-based ARQ, which uses the information from the Terminals to estimate the best retransmission strategy. In addition, the SDP-based ARQ offers a better protection of IDR packets in contrast with the rate-limited ARQ. While the rate-limited ARQ system assigns retransmission opportunities regardless of packet importance until it uses up the assigned retransmission rate, the SDP-based ARQ system prioritizes IDR packet retransmission.

In the event of no retransmission requests available at the Video Gateway, which results are detailed in Requests Off from Table 1, the SDP-based ARQ system keeps achieving the highest overall recovery rate and IDR recovery rate. When compared to the rate-limited ARQ system, the SDP-based ARQ offers a better protection of IDR packets even without information from the terminals.

This is due to the different treatment of packets according to their importance used in the SDP-based ARQ system. If there are no retransmission requests, the SDP-based ARQ system will retransmit those packets whose loss produces the highest expected distortion at the terminal. The rate-limited ARQ system, on the other hand, merely retransmits packets randomly, without taking their importance and their associated expected distortion into consideration.

It should also be noted that, while the rate-limited ARQ system may retransmit packets more than once, the SDP-based ARQ system achieves higher recovery rates with a single retransmission opportunity for each packet. Thus, our SDP-based ARQ reduces the congestive effect of retransmissions specially during error bursts while offering a higher protection.

#### 5. CONCLUSIONS AND FUTURE WORK

We have formulated video wireless transmission control as a SDP optimization problem, designing and implementing a wireless transmission system based on retransmission strategies. Our preliminary results, obtained in a real scenario of video distribution over 802.11g,

show that our SDP-based ARQ system performs better than a similarly rate-limited ARQ system without intelligent retransmission strategies.

Our proposed system is also better at protecting the most important information thanks to the retransmission policies, whose off-line calculation reduces the online complexity of our system. In addition, its computations for the online stage are reduced to state estimation, while the solution to the optimization problem is calculated off-line.

Future work in the SDP approach for wireless transmission includes an evaluation of our system for a broader variety of cases and an in-depth study and improvement of our system model and the optimization of a joint ARQ-FEC mechanism.

#### 6. REFERENCES

- [1] R. Schmitt and R.Y. Chen, "Wireless home entertainment networking using IEEE 802.11e QoS WLAN," in *Consumer Electronics, 2005. ICCE. 2005 Digest of Technical Papers. International Conference on*, Jan. 2005, pp. 305–306.
- [2] Hongqiang Zhai, Xiang Chen, and Yuguang Fang, "How well can the IEEE 802.11 wireless LAN support quality of service?," *Wireless Communications, IEEE Transactions on*, vol. 4, no. 6, pp. 3084–3094, Nov. 2005.
- [3] J. Chakareski and B. Girod, "Rate-distortion optimized video streaming with rich acknowledgments," in *Proc. Visual Communications and Image Processing*, 2004, pp. 661–668.
- [4] J. Chakareski and P. A. Chou, "RaDiO edge: rate-distortion optimized proxy-driven streaming from the network edge," *IEEE/ACM Trans. Netw.*, vol. 14, no. 6, pp. 1302–1312, 2006.
- [5] M.-H. Lu, P. Steenkiste, and T. Chen, "Video streaming over 802.11 WLAN with content-aware adaptive retry," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, July 2005, pp. 723–726.
- [6] S. Benayoune, N. Achir, K. Boussetta, and K. Chen, "Content-Aware ARQ for H.264 Streaming in UTRAN," in *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, 31 2008–April 3 2008, pp. 1956–1961.
- [7] D. P. Bertsekas, *Dynamic Programming*, Prentice Hall, 1987.
- [8] J. Cabrera, J.I. Ronda, A. Ortega, and N. Garcia, "Stochastic rate-control of interframe video coders for VBR channels," in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, Sept. 2003, vol. 3, pp. III–813–6 vol.2.
- [9] J. Cabrera, A. Ortega, and J.I. Ronda, "Stochastic rate-control of video coders for wireless channels," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 12, no. 6, pp. 496–510, Jun. 2002.