Adaptive Hybrid Error Control for IP-based Continuous Media Multicast Services*

Georg Carle, Henning Sanneck, and Mirko Schramm

GMD Fokus, Kaiserin-Augusta-Allee 31, D-10589 Berlin
carle,sanneck,schramm@fokus.gmd.de

Abstract. To overcome IP packet loss for continuous media applications, efficient error control mechanisms are required that take into account real-time constraints while trading off additional bandwidth and application level service quality. For multicast error control, heterogeneity caused by different receiver loss resiliency characteristics must be addressed as well. Related work has shown the advantages of hybrid error control that combines Forward Error Correction (FEC) with retransmission of parity packets (ARQ). We present a novel scheme for dynamically adapting hybrid error control to changing network conditions that minimizes the impact of packet loss on the received media quality. For the adjustment of the appropriate amount of redundancy an optimization mechanism using a set of equations which reflect network and media characteristics is applied. Analytical and simulation results are presented demonstrating the usefulness of the new scheme.

1 Introduction

In today's Internet there is an increasing amount of real-time multicast traffic for applications like teleconferencing, video distribution and distributed games. As IP services with QoS support (IntServ [1] and DiffServ [2]) are frequently either unavailable or costly, error control schemes are important that allow to use these applications with best effort IP services. Recent work [3] has demonstrated the advantages of retransmitting parity packets for efficient multicast error control. It has also been shown [4] that for real-time applications, *proactive* FEC can considerably reduce the required number of retransmissions for real-time applications. However, good solutions for adjusting the right amount of redundancy in changing network conditions are still missing: Too little redundancy may result in too many retransmission rounds needed for the loss recovery, while too much redundancy increases the loss rate of all flows over a common bottleneck. We propose an error control scheme for real-time multicast that allows to adjust and to optimize the number of parity packets for the initial transmission as well as for subsequent retransmission rounds. A set of equations for the adjustment of the amount of redundancy is presented that describes the adaptation to network and application characteristics. The remaining sections are organized as

* To appear in *First International Workshop on Quality for future Internet Services (QofIS 2000)*, Berlin, Germany, September 2000

follows: in section 2, the new scheme is introduced together with related work. Simulation results are presented in section 3. Section 4 concludes the paper.

2 Real-time Reliable Multicast Error Control

The majority of work on multicast error control focuses on non-real-time services. Huitema [5] studied the benefits of FEC when a separate retransmission scheme is used on top of the FEC layer. Nonnenmacher et al. compared a number of FEC/ARQ schemes in [3] and showed that an integrated scheme with ARQ of parities achieves the lowest network overhead since a fixed set of repair packets can recover a large variety of losses. Rubenstein et al. presented a repair technique called *proactive FEC* [4] which allows to decrease the expected time for a reliable receipt of data while achieving a required probability of successful delivery. Receivers should request more repair packets than they need to achieve a certain probability for a successful last retransmission or a certain probability for a successful reception in a maximum number of retransmission rounds. The drawback of this protocol is that it tries to fulfill all requests of the receivers at the cost of bandwidth which is not suitable for congested networks. Furthermore, the selection of a appropriate amount of proactive FEC is not mentioned. Kermode introduced a hierarchical protocol (SHARQFEC [6]) with the ability of localizing repair traffic and the selective addition of FEC. Though SHARQFEC is by nature a protocol for reliable multicast it introduces features which are also suitable for real-time multicast. The protocol uses a hierarchy of administratively scoped nested regions to restrict the range of repair traffic. Receivers are combined to repair zones based on subtrees. Each zone has a *zone closest receiver* (ZCR). For every repair zone the maximum local loss count defines the amount of redundancy added to this region by the ZCR. FEC can be selectively used in regions with higher losses. The amount of redundancy added for every region should depend also on the average loss and not only on the maximum loss rate like in SHARQFEC to improve the use of bandwidth. As regions may have largely varying loss rates to different receivers, adjusting the amount of redundancy to the receiver with the highest losses may be highly inefficient.

2.1 New Scheme: Adaptive Real-Time Scoped Hybrid ARQ FEC

In the following we present a new scheme for adaptive hybrid error control for real-time services. The scheme has been designed as an extension of SHARQ-FEC. The main differences to SHARQFEC are the support for real-time flows and the improved scheme to adjust proactive redundancy dynamically. Key features of the scheme are the following: Feedback consists of NAKs, information on the service quality ("service value") and the network conditions (like RTT and loss rate) of individual receivers. Both senders and receivers control the amount of proactive redundancy using service value functions. These functions allow to adapt the error control scheme according to the needs of receivers that value this most, as well as to the media characteristics and current network conditions. The scheme takes the negative impact of additional traffic on the loss rate into account. This is performed by adaptivity policies, which also reflect the network load conditions. The scheme complements TCP-friendly multicast rate adaptation by indicating which part of the consumed bandwidth would be used for FEC.

General Assumptions. After joining the multicast session, each participant receives the real-time flow via RTP/UDP on top of IP multicast from the sender. ADUs (Application Data Units) are transmitted as a transmission group combined with encoded proactive FEC (parity) packets to the so-called block (we employed Reed-Solomon Erasure Codes [7]). If the receiver does not get sufficient packets to decode the transmission group out of the received block it sends a NAK with the number of additional repair packets needed. This procedure is repeated until the receiver has sufficient packets for decoding or until the play-out point for the ADU has been reached. Beside this procedure every receiver has to generate its individual service value to evaluate the current quality of service by applying a user specified service value function. Every receiver periodically sends the service value, the play-out delay, the loss characteristics of the received packets and the RTT to the sender. Feedback of the receivers is used in two ways: a) to decide if the current service quality satisfies the one required by the receivers and b) to generate information necessary for optimization of the error correction while maintaining a high service quality. To decide if the parameters of the transmission should be modified, the sender uses the service value policy. This policy defines how the sender deals with unsatisfied receivers.

Service Value. For properly adjusting the error control scheme to application requirements and network conditions, the impact of packet losses on quality of continuous media services is important. For such an assessment we define the *service value* to be a number between 1 (lowest possible service value) and 5 (highest possible service value) like MOS (Mean Opinion Score) results of subjective tests [8].

Service Value Policy. The service value policy is used by the sender to decide if it should change the parameters of the transmission to increase or to reduce the quality at the receivers by influencing the loss rate. One has to remind that a single bad receiver can influence a whole group of good receivers if the service policy demands that all service values are over a certain level. Therefore, we decided to keep the deviation from the maximum value below a certain value for the whole group. For this policy the sender only needs to know the information of the unsatisfied receivers, for all other receivers it assumes that they are satisfied. This matches well with our scheme for the feedback regarding the scalability which depends on the group size and on the constraint that discontent receivers issue session messages much more often than satisfied ones.

Error Control. Improvement of error control is one way to increase the quality of real-time traffic. The main goal is to reduce the ADU loss rate at the receivers. Error control should allow to repair as many losses as needed to keep the service quality high. Two tasks are to be fulfilled: the first is to find all possible solutions in order to achieve a certain service quality for the receivers and the second is to choose the most effective solution that realizes this.

Bandwidth Adjustment. Reducing the ADU loss rate can be done by enlarging the bandwidth for error correction. There are three possibilities to obtain this additional bandwidth: adding it to the current bandwidth consumed, reducing the bandwidth used for the transmission of data or applying a combination of both. Since the number of available retransmission rounds is limited the only way to further enlarge the bandwidth for error correction is to increase the fraction of FEC (i.e. the amount of proactive redundancy). Many protocols and schemes exist that add proactive redundancy to the initial transmission ([6], [4]). However, since the maximum consumable bandwidth is often limited, such a limit should be determined by a separate mechanism which we do not consider further. We concentrate on the adjustment of a suitable ration of bandwidth for data transmission and error correction.

2.2 Service Value Equation

For the calculation of the service value equation we assume a star topology with randomly distributed losses on the links described by the loss probability p. The following variables are used: the number of data packets k, the number of parity packets h and the number of packets of the block n with $n = k + h$.

Retransmission Rounds. First the maximum number of retransmission rounds for which proactive redundancy could be adjusted has to be calculated. It is better to limit the number of retransmission rounds with adjustment of proactive redundancy depending on the characteristics of all receivers (especially if retransmissions are sent by multicast to all receivers). Otherwise, receivers with short RTTs would be favoured and moreover, the paths of high RTT receivers burdened. The number of available retransmission rounds with an adjustment of proactive redundancy (rr) should be calculated by dividing the mean playout-delay by the maximum RTT of all receivers. The complete transmission of one block consists of the initial transmission and up to rr retransmission rounds with adjustment of proactive redundancy. We number the rounds starting with round 0. In this round the initial transmission takes place and the receivers send their first NAKs. In each round ($r > 0$) the sender reacts to the feedback from the previous round. It issues as many repair packets as requested and additional repair packets equal to the current amount of proactive redundancy adjusted for this retransmission round.

The equation of the service value is composed of the following set of equations that reflect the influence of network conditions and media characteristics:

$$P\{round_r\} = \sum_{i=k_r}^{k_r+h_r} \binom{k_r+h_r}{i} (1-p)^i p^{k_r+h_r-i} \qquad with \quad 0 \le r \le rr \tag{1}$$

$$P\{transm.\} = 1 - \Big(1 - P\{round_0\}\Big)\Big(1 - P\{round_1\}\Big)\cdots\Big(1 - P\{round_{rr}\}\Big) \tag{2}$$

$$SV_{bw} = c_{dist} SV_{dist}\Big(P\{transm.\}\Big) + c_{noise} SV_{noise}\Big(P\{transm.\}\Big) \qquad (3)$$
$$with \quad c_{dist} + c_{noise} = 1$$

$$SV_{gaps} = c_1 SV_{bw}(1) + c_2 SV_{bw}(2) + \cdots \qquad with \quad \textstyle\sum_i c_i = 1 \qquad (4)$$

$$SV_{types} = c_A SV_{gaps}(A) + c_B SV_{gaps}(B) + \cdots \quad with\; c_A + c_B + \cdots = 1 \; (5)$$

Equation 1 represents the fundamental equation for the service value which is the calculation of the probability of a successful initial transmission or retransmission in any round. For each retransmission round the mean number of requested repair packets that is needed can be calculated ([9]) according to works of Rubenstein [4] and Nonnenmacher [3]. To compute the requested repair packets for the first retransmission round we use the variables $k_0 = k$ and $h_0 = h$. The calculated number equals the mean number of repair packets the sender has to issue in the first retransmission round and is defined as k_1. The number of proactive redundancy packets possibly added to the first retransmission is defined as h_1. With these two variables the sender can compute the mean number of requested repair packets for the second retransmission round which is defined as k_2. By repeating this procedure the sender can calculate numbers of requested repair packets for all retransmission rounds up to round rr. Taking the probabilities of a successful reception for all phases of the transmission for one block we get the equation 2 for the probability of success for the complete transmission. Consequently, the ADU loss probability p_{ADU} can be estimated by: $p_{ADU} = 1 - P\{transm.\}$. For every combination of $h_0, h_1, \ldots h_{rr}$ another equation will be built because if h_r increases the number of requested repair packets for the following retransmission round, k_{r+1} decreases and vice versa. Equation 3 represents the impact of bandwidth utilization. The more bandwidth is allocated for error correction, the smaller the ADU loss rate (p_{ADU}) is. We defined this service value as SV_{dist} since typically lost ADUs are perceptible by temporary "distortion". However, since the maximum available bandwidth is often limited, the bandwidth for data transmission is also reduced. Obviously, less data bandwidth will reduce the perceived quality by continous "noise". Therefore, we defined this service value as SV_{noise}. The service value for bandwidth should be a combination of these two components. The components are both weighted with a factor representing their relative importance. For instance, using waveform-coded audio, lost ADUs lead to distortions caused by signal interruptions [10, 11] while the available data bandwidth is perceived by the amount of quantization noise (assuming that the data bandwidth is varied by adjusting the bit resolution of the audio signal [12]). The influence of loss distribution can be captured by equation 4. While some single missing ADUs can be tolerated up to a certain degree, a gap of consecutive ADUs typically leads to a higher loss impact (assuming the same mean loss rate for both cases) depending on the coding scheme. Therefore, proactive redundancy should be adjusted over successive transmission groups to be able to avoid this. Finally, equation 5 can be applied

equation for	range	influenced by
$P\{round\}$	one round	mean loss
$P\{transm.\}$	transmission group	RTT
SV_{bw}	one ADU	bandwidth
SV_{gaps}	consecutive ADUs	mean loss gap
SV_{types}	different ADUs	media

Table 1. Ranges for the service value equations

if there are ADU types with different importances for the service quality (like different frame types in video).The specific utility functions (for instance for loss gaps or frame types) are included as expressions for SV_{dist} in the respective SV equation for the corresponding loss gap or frame type. In Table 1 the ranges for the different equations can be seen together with the parameters that influence them.

With this set of equations, a single equation for the service value can be generated for specific network conditions and media characteristics. By using this equation for the adjustment and the optimization of the amount and distribution of the proactive redundancy, the error control can be adapted to special conditions and thus can work in an improved way for the service.

Service Value Example. For illustration of the new measure we give a simple example. We assume that bandwidth limitations allow five packets (which consist of either data or redundancy) to be used for transmitting each ADU of a continuous media application. The quantity of redundancy packets determines the ability for error correction. In our example the highest possible data bandwidth is five, i.e. all five packets can consist of data. If data packets are replaced by redundancy packets on the one hand the bandwidth consumed for data is decreased but on the other hand the ADU loss rate is reduced too.

With no losses the service value is very high if little or no redundancy is used. In this case the service value heavily decreases with higher loss rates. If more redundancy is used the service value is lower for low loss rates but remains unaffected for higher losses. We obtain the following formula for representing the tradeoff:

$$SV = c_{dist}(5 - 4p_{ADU}) + c_{noise}(4/bw_{data_{max}} bw_{data}(1 - p_{ADU}) + 1)$$

The service value with equal factors c can be seen in figure 1. By modifying the factors one can alter the graphs in the following way: if on one hand $c_{dist} > c_{noise}$ the graphs will move upwards and thus more redundancy will be chosen with lower loss rates and the service becomes more reliable. If on the other hand $c_{dist} < c_{noise}$ they move downwards and more redundancy and thus fewer data packets will be chosen only with higher loss rates and the service tries to keep a high quality but with some interruptions.

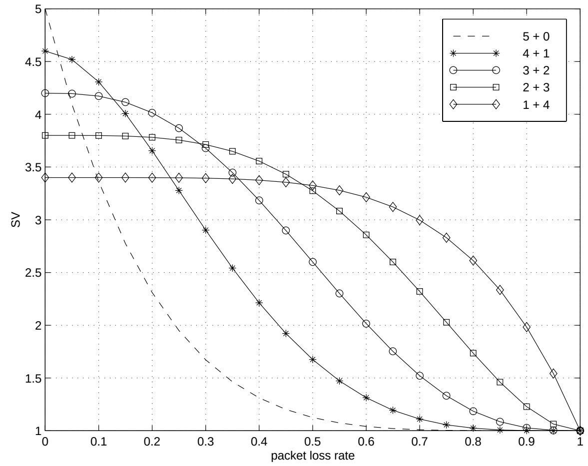

Fig. 1. SV for different redundancies and packet loss rates with $c_{dist} = c_{noise} = 0.5$.

2.3 Optimization of Proactivity

Having many possibilities to change the distribution of the proactive redundancy for subsequent transmission rounds, a mechanism is needed to choose the most suitable solution. Such a mechanism has to calculate the probability of success (service value) for all possible amounts and distributions of the proactive redundancy by varying $h_i \ \forall \ i \in [0, rr]$. A limit has to be defined for the maximum amount of proactive redundancy for each round. We limited h_i to the same value as for k_i, that is 100% proactive redundancy for each round. To be able to choose the most suitable distribution of proactive redundancy the mechanism needs optimization criteria. A simple but very effective criterion is the bandwidth consumed. For its estimation we used the case of multicast retransmissions. While k_0 and h_0 are always sent, k_1 and h_1 are issued only with the probability $1 - P\{round_0\}$ (i.e. the probability for a failed first transmission), a.s.o. Therefore, the average number of packets necessary for one transmission group can be calculated as follows (an example can be found in [9]):

$$packets = k_0 + h_0 + \sum_{i=1}^{rr} (k_i + h_i) \prod_{j=0}^{i-1} (1 - P\{round_j\})$$

The bandwidth consumed for the transmission of data is expressed by the variable k_0. The other part of this equation expresses the bandwidth used for error correction with all k_i $(i \geq 1)$ representing the portion of reactive redundancy (ARQ) and all h_i $(i \geq 0)$ representing the portion of proactive redundancy (FEC). A new adjustment of proactive redundancy takes place if the service policy demands an improvement of the service value. The goal is to reduce the loss rate of the ADUs at the receivers. For that it is necessary to increase the bandwidth for the error correction. Obviously, the more proactive redundancy is added, the higher the probability of success and thus the lower the ADU loss rate. Nevertheless, with larger amounts of proactive redundancy (especially in the initial transmission) the difference in the resulting probability of success is getting smaller and the cost (in terms of bandwidth consumption) higher. Therefore, the gain of a successful transmission always has to be looked at in connection with the necessary bandwidth. One can distinguish the following principles for an optimization (figure 2), which is to find one of the following:

a) the smallest bandwidth to achieve a constant service value
b) the highest ratio of the service value and the consumed bandwidth
c) the highest service value with a constant bandwidth

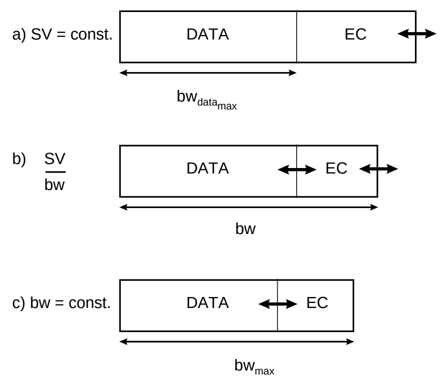

Fig. 2. The principles for optimization regarding the bandwidth.

The first principle should be used if the network load is low (and thus error control is used to ensure very low ADU loss rates) and the bandwidth enlargement leads to almost no increase of the packet loss rate. In such a situation the main target is to keep the service value as constant as possible at the demanded high value. If the network load is neither very high nor very low an increase of bandwidth for the error correction should depend on the gain in terms of achieved service quality. Therefore, the service quality should be related to the bandwidth consumed for it which is represented by principle b). Principle c) should be used in cases of high network load. A further enlargement of the bandwidth leads with a high probability to an increase of the packet loss rate. This reduces the service quality not only for the receivers but also for the other traffic sharing the same links. Therefore, the bandwidth should be kept constant and might even be reduced by a bandwidth adaptation mechanism (c.f. TCP-friendly adaptation schemes like in [13]). Furthermore, flows that do not reduce their bandwidth might be penalized by routers too [14], so that the service quality of the receivers decreases further. The decision which of these three possibilities to choose depends on the traffic mix and on the bandwidth used regarding the bottleneck bandwidth.

3 Simulation

The simulations were carried out using the network simulator *ns-2* [15]. We set most of the parameters as in [6], using a simple topology with four receivers. The sender emitted 200 ADUs, each consisting of 8 data packets, at a rate of 800 kbit/s. We wanted to test the schemes with a heterogeneous scenario and therefore, we set the loss rates of each link to either 0.05 or 0.2 and the RTTs to either 60 or 100 msec. The play-out delay was adjusted to 550 msec. Since only the real-time flow was simulated, a loss model for every link has been used to cause losses in the flow. We used two measures to be able to compare the performance of the different schemes: the mean deviation and the mean quadratic deviation of the service value from the maximum (which is always 5). The quality is optimal if all receivers have a service value of five over the whole simulation time. Therefore, the difference between the current service value and the optimum value, respectively its square, was added for each receiver and divided by the time. Finally, the mean value of all receivers was calculated. Furthermore, we looked at the amount of issued packets (data and repair packets) to be able to compare the effort to realize this deviation. Since this quantity strongly vacillates we introduced an average over the last 20 transmission groups.

3.1 Variation of Play-out Delay

Since the play-out delay together with the current RTT mainly define the available number of retransmission rounds they influence the success of several error correction schemes. Therefore, we first investigated the dependency of the deviation of the service value on various play-out delays. In our test topology the RTT is either 60 msec or 100 msec. The NAK suppression delay varies between 2 and 2.5 times of the time estimation for the distance to the sender (1/2 RTT). We compared the original SHARQFEC mechanism with our scheme. In figure 3 the relative service quality deviation can be seen. Each graph has nearly the same form which is explainable by the number of retransmission rounds that can be used. If the play-out delay is very small (below 400 msec) there are only one or two retransmission rounds available for the receivers with small RTTs and even none for the others. Therefore, the deviation strongly decreases with higher play-out delays. If the play-out delays grows higher than 400 msec the deviation is further reduced since more retransmission rounds are usable. Especially if there is only a small play-out delay (and thus only a few or no retransmission rounds at all) available our scheme leads to a smaller deviation and thus to a better service quality. The amount of proactive redundancy is adjusted in such a way that more losses can be recovered with the available retransmission rounds.

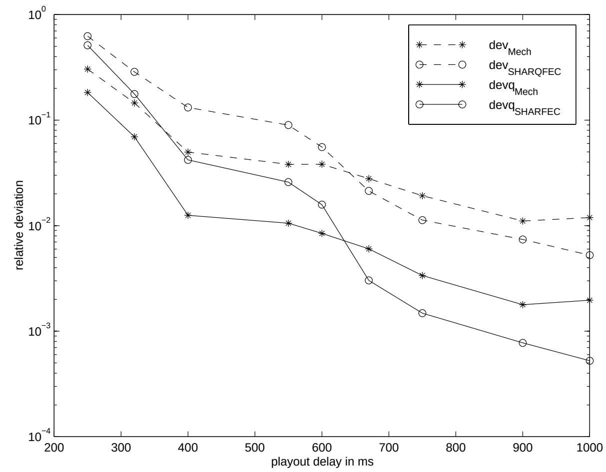

Fig. 3. Service quality deviation depending on the available play-out delay.

As our scheme adjusts only redundancy if the service policy decides so, the deviation is slightly higher with long play-out delays (more than 650 msec). Since only a few or even no adjustments have been done only a very small amount or even no proactive redundancy is added. This is because the majority of losses can be repaired completely by retransmission alone. Nevertheless, by modifying the tolerated deviation from the maximum value (which results in other service policies) the deviation of the service quality can be reduced even for long play-out delays. For all other simulations the play-out delay was set to 550 msec to allow some retransmission rounds (at least two for the receivers with high RTTs) if not otherwise stated.

3.2 Variation of Loss

The next property investigated was the dependency of the service quality deviation on a varying packet loss rate. For this purpose, we modified the loss rates at the links of our test topology between 0.01 and 0.2 (the loss rate was the same at all links). Figure 4 shows again a comparison of the original SHARQFEC mechanism with our scheme, which adjusts the redundancy to all retransmission rounds. Our scheme performs better than SHARQFEC for all investigated loss rates because the amount of proactive redundancy is adjusted in such a way that more losses can be recovered with the available retransmission rounds.

3.3 Error Control Mechanisms

In this section we compare several schemes to adjust proactive redundancy. As a reference we took SHARQFEC which changes only the amount of proactive

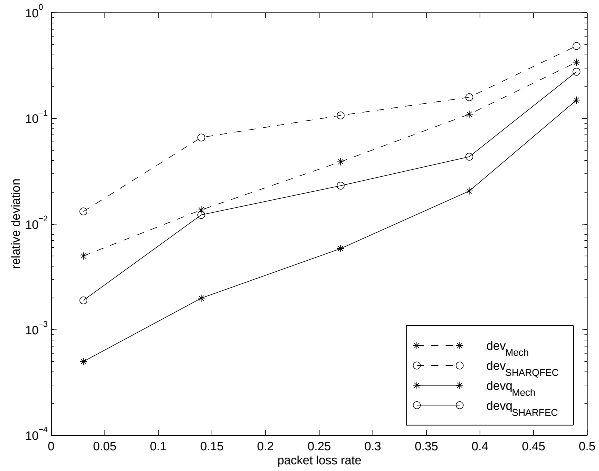

Fig. 4. Service quality deviation depending on the packet loss rate.

redundancy for the initial transmission (h_0) depending only on the zone loss count of the previously sent blocks (figure 5: we applied SHARQFEC without scoping and injection of redundancy by receivers). Although the amount

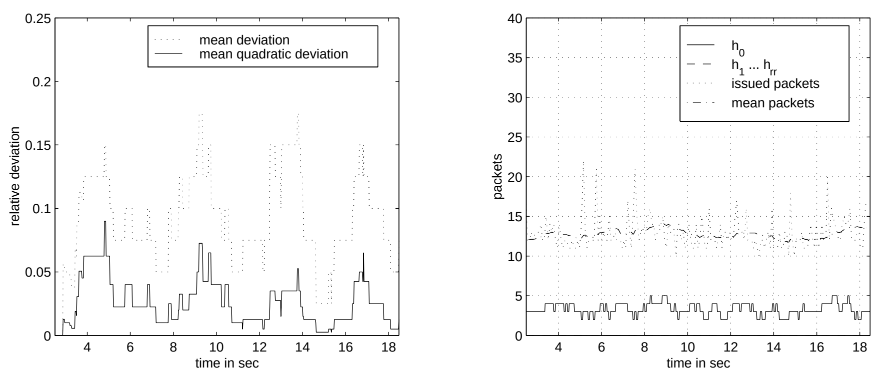

Fig. 5. Service quality deviation and issued packets of SHARQFEC.

of proactive redundancy alters between 2 and 5 packets (with a mean value of slightly more than 3 packets) the service quality deviation can be only partly reduced. This is because proactive redundancy mainly depends on the short-term measure zone loss count and therefore, it is not sufficient for blocks with higher losses if the previous sent blocks suffered only smaller losses. Consequently, a long-term adjustment should be applied. Therefore, we evaluated a scheme with an alteration of h_0. It is similar to the idea applied by Rubenstein ([4]). The mechanism calculates the probability of success for every possible number of proactive redundancy packets and chooses the solution with a minimum sufficient probability (figure 6). The results for that scheme are much better because

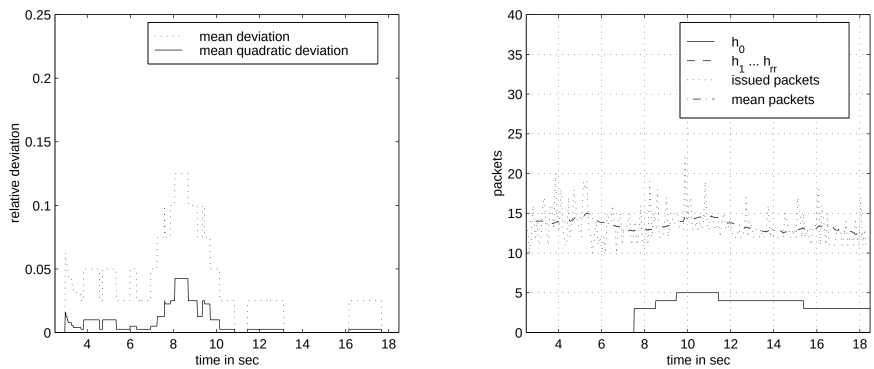

Fig. 6. Service quality deviation and issued packets of a mechanism which optimizes h_0.

just in the first adjustment h_0 was set to 3 which leads to a much shorter period of the service quality degradation. In the next two adjustments the amount of proactive redundancy is stepwise increased which results in a service quality deviation close to zero. However, it should be possible to reduce h_0 somewhat by adding some proactive redundancy to the retransmission rounds. Therefore, we tested our scheme with an adjustment of all h_i values. (figure 7). Our scheme reaches a similar service quality deviation compared with the sole adjustment of h_0. Furthermore, fewer proactive redundancy packets were adjusted for the initial transmission. On the one hand, this results in fewer packets sent in the initial transmission but on the other hand, the amount of packets issued during the retransmissions can be larger (this is shown in figure 7 if one compares the smallest and the highest values of the issued packets). To avoid the degradation of the service value until a suitable amount of proactive redundancy is adjusted we combined our scheme with SHARQFEC which changes h_0 depending only on a short-term measure (figure 8: in the figure only the amount of proactive redundancy adjusted by our scheme can be seen). The gain of such a combined mechanism is significant because the deviation of the service quality is the lowest of all investigated schemes for the adjustment of proactive redundancy. This is because the short-term adjustment of SHARQFEC avoids strong degradations of the service quality (especially at the beginning of the session or if the packet loss rate strongly changes) while our scheme adjusts a certain amount of proactive redundancy which realizes a constant service quality.

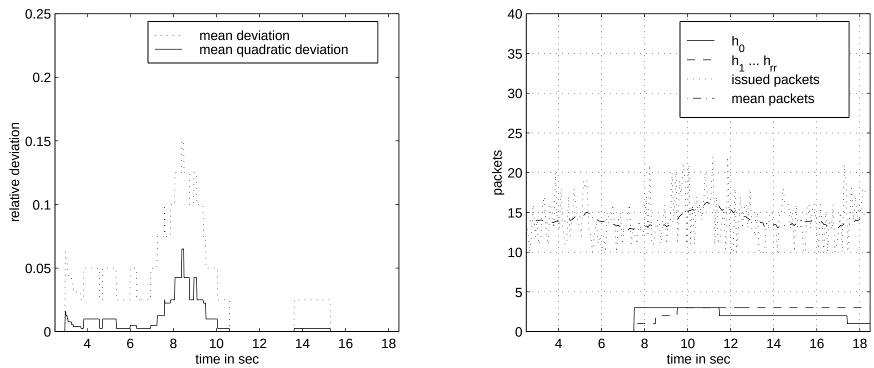

Fig. 7. Service quality deviation and issued packets of a mechanism which optimizes all h_i.

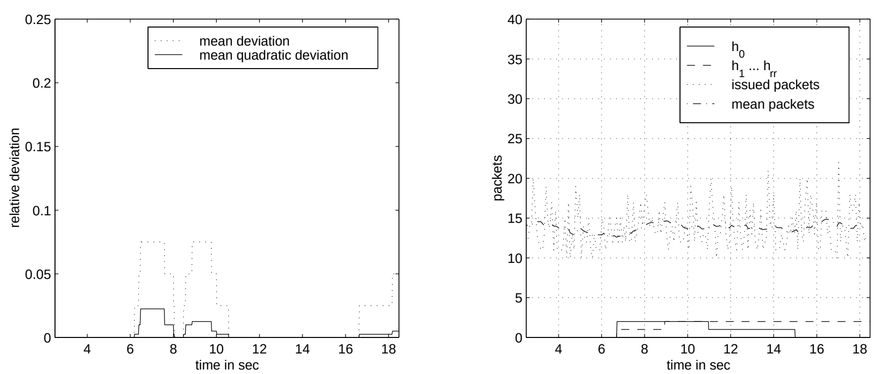

Fig. 8. Service quality deviation and issued packets of a combination of SHARQFEC and optimization.

4 Conclusions

For error control of multicast transmissions recent work showed that a combination of FEC- and ARQ-methods is most effective. The new aspect of this work is the distribution of additional proactive redundancy for the retransmission rounds, too. Therefore, it is possible to transmit with a higher probability of success and thus further reduce loss and delay. A new measure - the service value - was introduced to offer a possibility to evaluate the influence of losses on real-time streams depending on the bandwidth, the loss distribution and the transmitted media characteristics. Using this new measure a set of equations was generated which is able to adapt the error control strategy to the service value. A single equation is built from this set and then applied by the scheme for the adjustment and the optimization of the amount and distribution of the proactive redundancy. Combined with SHARQFEC our scheme leads to a significant increase of the service quality since the proactive redundancy is influenced in two ways: The short-term adjustment of SHARQFEC avoids significant degradations of the service quality due to changing network conditions. The long-term adjustment of our scheme allows an adaptation of the error control to the specific network conditions and usage scenarios.

References

1. R. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture: an Overview," RFC 1633, IETF, 1994.
2. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wan, and W. Weiss, "An Architecture for Differentiated Services," RFC 2475, IETF, 1998.
3. Jörg Nonnenmacher, Ernst W. Biersack, and Don Towsley, "Parity-Based Loss Recovery for Reliable Multicast Transmission," in *Proceedings of ACM SIGCOMM'97*, Cannes, France, September 1997.
4. Dan Rubenstein, Jim Kurose, and Don Towsley, "Real-Time Reliable Multicast Using Proactive Forward Error Correction," Technical Report 98-19, Department of Computer Science, University of Massachusetts, March 1998.
5. Christian Huitema, "The Case for Packet Level FEC," in *Proceedings of IFIP 5th International Workshop on Protocols for High Speed Networks (PfHSN'96)*, INRIA, Sophia Antipolis, FRANCE, October 1996, IFIP, Chapman & Hall.
6. Roger G. Kermode, "Scoped Hybrid Automatic Repeat reQuest with Forward Error Correction (SHARQFEC)," in *Proceedings of the ACM SIGCOMM Conference*, Vancouver, B.C., September 1998.
7. L. Rizzo, "Effective erasure codes for reliable computer communication protocols," *Computer Communication Review, ACM SIGCOMM*, April 1997.
8. John G. Gruber and Leo Strawczynski, "Subjective Effects of Variable Delay and Speech Clipping in Dynamically Managed Voice Systems," Tech. Rep., IEEE Transactions on Communications, August 1985.
9. M. Schramm, "Multicast capable error control mechanisms for IP-based multimedia services," Diploma Thesis, GMD Fokus / Technical University of Berlin, Berlin, July 1999.

10. H. Sanneck, "Concealment of lost speech packets using adaptive packetization," in *Proceedings IEEE Multimedia Systems*, Austin, TX, June 1998, pp. 140–149, ftp://ftp.fokus.gmd.de/pub/glone/papers/Sann9806:Adaptive.ps.gz.
11. J.-C. Bolot, S. Fosse-Parisis, and D. Towsley, "Adaptive FEC-based error control for interactive audio in the Internet," in *Proceedings IEEE INFOCOM*, New York, NY, March 1999.
12. T. Turletti, S. Fosse-Parisis, and J.-C. Bolot, "Experiments with a layered transmission scheme over the Internet," Research Report 3296, INRIA, November 1997.
13. D. Sisalem and A. Wolisz, "LDA+ TCP-friendly adaptation: A measurement and comparison study," in *Proceedings NOSSDAV*, Chapel Hill, NC, June 2000.
14. R. Guerin and V.Peris, "Quality-of-Service in Packet Networks: Basic Mechanism and Directions," Research report, IBM Research Division, 1997.
15. UCB/LBNL/VINT, *Network simulator ns-2*, October 1998, http://www-mash.cs.berkeley.edu/ns/ns.html.
16. Jean-Chrysostome Bolot, "Characterizing End-to-End Packet Delay and Loss in the Internet," *Journal of High-Speed Networks*, vol. 2, no. 3, pp. 305–323, December 1993.