Performance of IP Header Compression over Correlated Multiple Channels

Tatiana Kozlova Madsen, Frank H.P. Fitzek, Yasushi Takatori, Ramjee Prasad
Department of Communications Technology, Aalborg University
Neils Jernes Vej 12, 9220 Aalborg Øst, Denmark,
e-mail: *[tatiana|ff|yt|prasad]@kom.auc.dk*

***Abstract*— Exploitation of multiple parallel channels is one of the recent trend in wireless communications. Splitting the connection into the multiple flows offers the possibility to obtain diversity gain by transmitting via different paths. Another trend is the demand for wireless delivery of data services. For efficient usage of bandwidth, IP header compression can be applied. In this paper we investigate the performance of IP header compression techniques over correlated parallel channels and show that the cooperative behavior between different channels can result in performance gain. Furthermore, the tradeoff between the compression gain and robustness of the compression algorithms is discussed.**

I. INTRODUCTION

The wireless market is believed to be very important for the future data services. As new services emerge for wired networks, there arises a user demand to incorporate those services in wireless communication systems. Despite the efforts that were made, the wireless protocol domain is still challenging. IP-based multimedia applications, including audio- and video-streaming and gaming, require more bandwidth than traditional voice services in circuit-switched networks. To use the limited resource, bandwidth, in the most efficient way, multimedia payload should be compressed and the IP header overhead should be reduced when possible. IP header compression mechanisms are an important part of saving bandwidth over bandwidth-limited links.

Another trend of communication development is exploitation of multiple channels. Compared with a single-channel communication, communication using multiple channels allows more flexibility, robustness and higher capacity. Examples of multiple channel communication can be found in both wired and wireless networks. Figure 2 presents a possible 4G scenario when a mobile terminal establishes several parallel connections with an access point (cellular or WLAN). Actually, the need for multiple channel communication can be dictated by applications. Novel applications support splitting of one stream of data into substreams. For example, downloading a web site, different TCP connections (for text, pictures etc) are established. Another example is usage of Multiple Description Coding (MDC) or Multi Layerd Coding (MLC) for coding of a video stream. Splitting the connection into the multiple flows offers the possibility to obtain diversity gain by transmitting via different paths.

In this paper we consider the performance of IP header compression schemes over parallel channels. Generally, header compression (HC) is possible due to redundancy among the header fields of a packet flow [1]. On the sender side, the compressor removes redundancy from the incoming packet using information from the past packets, called the *context*. The decompressor maintain the context and uses it to reconstruct the header of the incoming packet. The inconsistencies in the contexts of compressor and decompressor lead to the loss in synchronization and failure of the decompression procedure. Desynchronization occurs due to the losses of IP packets. Different IP header compression schemes are designed to withstand some amount of channel errors. But as we show in this paper not only the amount of erroneous packets, but their distribution affects the performance of compression algorithms.

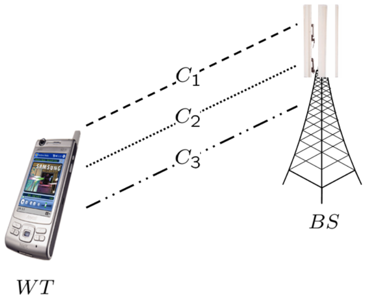

Fig. 1. Example of parallel channels in 4G network.

The paper is organized as follows. Section II provides some insights in header compression approaches with the emphasis on cooperative header compression. Section III presents the method for modelling the correlation among multiple parallel channels. Section IV shows performance evaluation in terms of packet error probability, bandwidth efficiency and network overhead. Section V gives some concluding remarks.

II. HEADER COMPRESSION FOR WIRELESS COMMUNICATION

Most of the HC schemes are designed for single-channel communication. Schemes like Compressed RTP (CRTP) [2] compress the redundant information within the packet headers. Analysis of the variations on the field information of the packet flow is used to decide the smallest amount of information needed to reconstruct the header fields on the receiver sizde. This information is mostly coded as delta values. Compressed headers are comprised of delta values and they are transmitted instead of the original header.

When multiple pathes are available between a source and a destination, Cooperative header compression (COHC) can be applied. This scheme proposes to send extra compressed information (additional information container - AIC) to minimize the desynchronization of compression contexts through the other links in piggyback form (see Figure 2). This scheme does not use the feedback from the receiver. The feedback is usually used to report the loss of the context. In case the feedback is unavailable or it is undesirable to use it, synchronization between the compressor and decompressor is maintained by periodic context updates. The rate of the updates depends on the channel conditions: if the channel is good, packets with the uncompressed headers should be sent less often. Additionally to periodic updates, COHC maintains the synchronization by using extra information provided by the neighboring channels. Generally, feedback is difficult or at all impossible to provide in cases of broadcast/ multicast and for delay-sensitive applications. The feedback should be received in proper time and it should be error-free, otherwise the compression scheme will be damaged. Using feedback can be spectrum-inefficient and if the feedback is constantly sent, the power consumption of a mobile terminal is increased. For these reasons, we have chosen to focus on schemes that do not rely on the feedback information.

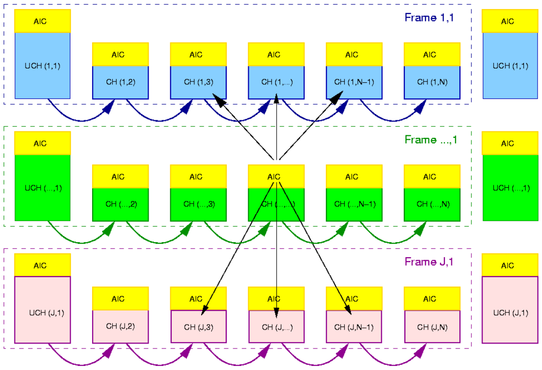

Fig. 2. Cooperative header compression approach.

A detailed description of COHC can be found in [3]. Since an AIC for a particular channel is carried by other channels, the performance of the scheme increases significantly due to the path diversity effect. In [4] it is shown that cooperative behavior results in performance gain in case of random error distribution. In this paper we investigate if there is any benefits of using cooperation between multiple channels in case of correlated channel behavior. We consider a HC scheme based on delta coding compression and an enhancement of this scheme by employing cooperative behavior.

III. MODELLING THE CHANNEL CORRELATION

For our analysis we consider synchronous parallel channels. By synchronization between multiple channels we understand synchronization of packet arrivals on different channels within the granularity of IP packets. In other words, IP packets that are transmitted over parallel channels will be received simultaneously. Re-synchronization can occur due to the different channel capacity and retransmissions at the link layer. The re-synchronization effect can be minimized by introduction of a buffer at the receiver side.

We propose the following approach to model the correlation between the channels. Figure 3 illustrates our method that can be interpreted as a Markov chain model. First we produce basic channel quality: errors are independently and identically distributed and q is the probability to loose an IP packet. Then we derive error distribution for each particular channel by defining transition probabilities: $p_i(0 \rightarrow 0)$ defines the probability that a packet is received correctly on channel i given that basic channel quality is good, and $p_i(1 \rightarrow 1)$ gives the probability to loose a packet given that the channel quality is bad. The advantage of this approach is that the error distribution on each particular channel is random, but varying the values for transition probabilities, we can influence the correlation between the channel behavior. If the transition probabilities $p_i(0 \rightarrow 0) = p_i(1 \rightarrow 1) = 1$ for all i, then the error patterns on different channels are the same and the channels are completely correlated. If e.g. $p_1(0 \rightarrow 0) = p_1(1 \rightarrow 1) = 1$ and $p_2(1 \rightarrow 1) = 0$, then the channels 1 and 2 are uncorrelated, since if there is an error on one channel there is no error on the other.

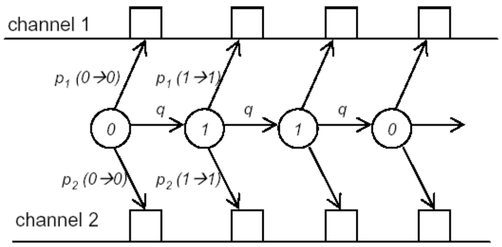

Fig. 3. Channel correlation as Markov chain

We can introduce a *correlation factor*, a parameter that represents an amount of correlation between different parallel channels. Correlation factor shows in how many cases an occurrence of a packet error on one channel will lead to the packet loses on other channels as well. It can be written as the following conditional probability:

$$C = Pr(errors\ on\ channels\ j \neq i \mid error\ on\ channel\ i)$$

Using our model, it can be found that

$$C = q \prod_{i=1}^{J} p_i(1 \rightarrow 1) + (1-q) \prod_{i=1}^{J} (1 - p_i(0 \rightarrow 0)) \qquad (1)$$

where J is the number of parallel channels. The correlation factor equal to one indicates a complete correlation between the channels; a completely uncorrelated situation corresponds to $C = 0$.

Another advantage of the described above approach is that it can be easily modified if there is a need to model bursty error distribution over time. In order to do that, a two-state Markov chain can be applied to model the good and bad states for the basic channel quality. Then the error distribution on the channels will also follow Markov chain model, though with different values for the transition probabilities and the average state holding time. In this paper, we limit our consideration only to the case of independently distributed errors.

IV. PERFORMANCE EVALUATION

In order to evaluate the cooperative approach for header compression, in this section we analyze its performance in terms of packet error probability (PEP), bandwidth efficiency (BE) and network overhead.

A. *Packet error probability*

Evaluating packet error probability, we take into account both losses of packets due to the channel condition and decompression errors. Considering the error model described above, the following formulas are derived for PEP:

$$PEP_{no\ cooper} = 1 - \sum_{i=1}^{J} \frac{(1-p_i)\left(1-(1-p_i)^N\right)}{JNp_i}$$

$$PEP_{with\ cooper} = 1 - \frac{\left(1-\frac{1}{J}\sum_{i=1}^{J} p_i\right)\left(1-(1-p_{SL})^N\right)}{Np_{SL}}$$

where p_i is the probability to loose a packet on channel i, p_{SL} is the probability of loosing synchronization between the compressor and decompressor and N is the number of packets in a frame. p_i can be found as

$$p_i = qp_i(1 \rightarrow 1) + (1-q)(1-p_i(0 \rightarrow 0))$$

Due to the cooperative behavior of the parallel channels, the synchronization between the IP header compressor and decompressor will be maintained in the case of a packet loss if a packet on at least one channel is received correctly. Loss of synchronization will occur only in the case when we have simultaneous loss of packets on all channels. Thus, p_{SL} coincides with the value of the correlation factor and it can be calculated by using formula (1).

One can notice that if the correlation factor is very small, then p_{SL} tends to zero, and using approximation $(1-p_{SL})^N = 1 - Np_{SL} + O(p_{SL})$ that is valid for small p_{SL} we get

$$PEP_{with\ cooper} = \frac{1}{J}\sum_{i=1}^{J} p_i$$

That is, the packet error probability is just an average of channel error probabilities of different channels. This is in correspondence with the intuition that if the correlation factor is very small, we always can recover the compression base and in this case the observed errors are only due to the channel behavior.

Figure 4 shows the PEP as a function of the channel quality in case when the channels are completely uncorrelated. In this case cooperation between channels can help to avoid desynchronization and PEP for the scheme with cooperation is significantly lower compared with the conventional scheme. One should note that Figure 4 will look the same for any number of channels: for independent channels we are taking an average of the error rates and for cooperative channels adding more channels will not result in the PEP improvement. In case the correlation between the channels exists, the number of cooperative channels will have a big impact on the PEP and the bigger the number of channels is, the lower is the error rate. In Figure 5 PEP is presented for the case of two correlated channels. One can still observe the advantage of using COHC. For Figure 5 the correlation factor is a function of the channel quality, and for each point in can be calculated as $0.2 - 0.1q$. It is approx. equal to 0.2 for the whole range of the considered values q. For comparison purpose, we have also presented the line corresponding to the uncompressed case. This line presents the lower bound for the error rate.

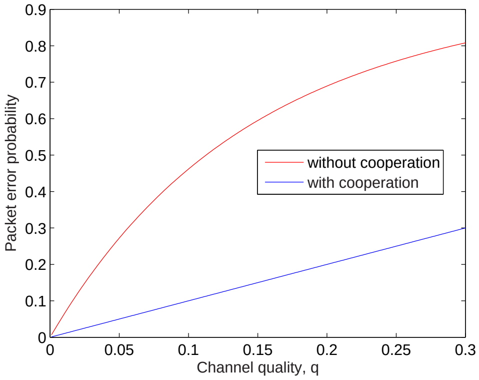

Fig. 4. PEP as a function of the channel quality. All transition probabilities are equal to 1.

Only in the case when the channels behavior is highly correlated, introducing cooperation will not result in the performance improvement. To achieve the performance gain we should "break" the correlation between the channels. One way of doing it is by using interleaving. By interleaving process, a burst of loss on parallel channels can be transformed into isolated losses. We would like to point out that instead of interleaving the whole IP packets, we propose to interleave only AICs. This will help to keep the latency low when no errors occur.

B. *Bandwidth efficiency*

Another important characteristic is bandwidth efficiency that reflects the tradeoff between the robustness of the scheme against packet losses and the compression gain. From one hand, we would like to reduce the size of a header as

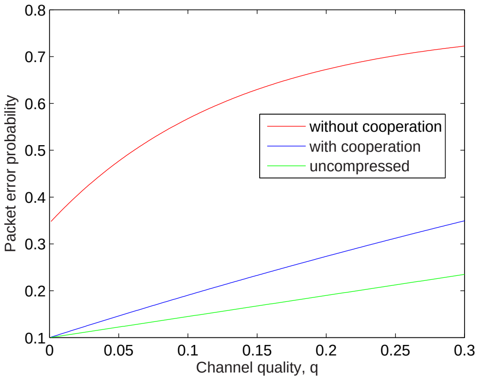

Fig. 5. PEP as a function of the channel quality. $p_1(0 \rightarrow 0) = p_2(1 \rightarrow 1) = 1, p_2(1 \rightarrow 1) = 0.1, p_2(0 \rightarrow 0) = 0.8$.

much as possible (high compression gain), from another hand, this information should be enough to keep the synchronization between the compressor and decompressor (robustness). Bandwidth efficiency shows how efficiently we are using the available bandwidth resources. It is defines as the ratio of the correctly received and decompressed useful information over the total amount of the information sent.

In this paper we are considering a header compression scheme that is relying on the cooperation to maintain the context synchronization without using a feedback from the receiver. To provide a reliable feedback in a wireless environment, additional signalling is required. By using cooperation, we can avoid excessive signalling, and thus, use the limited resource, bandwidth, in the optimal way.

In order to illustrate this point, we consider a hypothetical scheme with an ideal feedback. We assume that each time a packet is lost, a receiver provides a feedback and already the next packet is transmitted with an uncompressed header and the base at the receiver side is updated. We refer to this scheme as ”ideal”, since in a realistic scenario the feedback will not be received immediately and the sender will continue transmitting packets with compressed headers even if the base is lost. Nevertheless, it is interesting to compare the scheme with ideal feedback with cooperative header compression.

First, we estimate the average frame length of the ideal scheme. In a frame all packets are received correctly except the last one. Therefore,

$$N_{ideal} = \sum_{k=1}^{\infty} k(1-p)^{k-1} p = \frac{1}{p} \qquad (2)$$

Formula 2 supports our intuition: the smaller is the probability to loose a packet, the longer is the frame. Bandwidth efficiency can be written as

$$BE_{ideal} = \frac{D(N_{ideal} - 1)}{(DN_{ideal} + B_u + B_c(N_{ideal} - 1))}$$

where N_{ideal} is defined as in formula (2), D, B_u and B_c are the sizes of the payload, uncompressed and compressed headers, respectively.

Analogously, we define bandwidth efficiency for COHC:

$$BE_{COHC} = \frac{ND(1 - PEP)}{(ND + B_u + B_c(N-1) + N(J-1)B_a)}$$

where B_a is a size of one AIC.

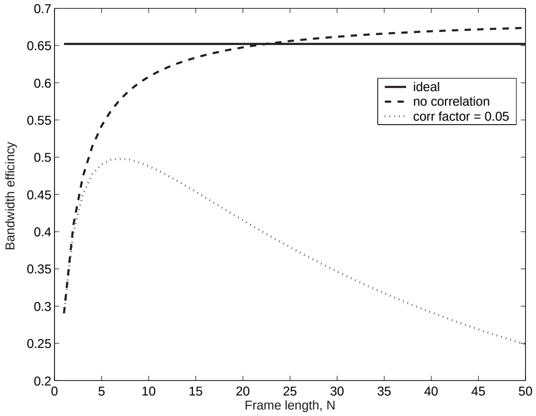

Fig. 6. Bandwidth efficiency as a function of the frame length. $D = 20$ bytes, $B_u = 40$, $B_c = 4$, $B_a = 2$ bytes, $J = 2$, $p = 0.1$.

Figure 6 shows bandwidth efficiency as a function of the frame length, N. Since the frame length of the ideal scheme varies, we present the average BE as a constant line. For COHC bandwidth efficiency is shown for two cases: completely uncorrelated channels and correlated channels (correlation factor is equal to 0.05). As expected, BE of the ideal scheme is high compared with the compression without using feedback, and the line for COHC in case of correlated parallel channels lies below the straight line for ideal scheme. But in the case of uncorrelated channels, BE of COHC increases with increase of N and when $N > 20$, it outperforms the ideal scheme. Indeed, when N is small, cooperative header compression sends frequent updates (packets with uncompressed headers); this reduces the overall efficiency of the scheme. Now lets consider the case of a long frame. With each packet COHC uses 4 bytes of a compressed header plus 2 bytes of an AIC compered with only 4 bytes of a header for the ideal scheme, that is, COHC sends bigger headers. But when a packet is lost, in order to recover the base the ideal scheme sends an uncompressed header of 40 bytes, whereas COHC can update the base without sending any extra bytes. Therefore, the cooperative scheme without feedback outperforms the scheme with ideal feedback in case of uncorrelated channel errors.

C. Network overhead

Finally, we would like to show that IP header compression reduces the network overhead significantly. We have chosen to use an example of Multiple Description Coding (MDC), an attractive coding scheme that has received a lot of interest lately [5]. MDC can efficiently cope with bursty error characteristics typical for wireless environment and provide robust transmission. It is achieved by splitting one source of information into multiple streams, called descriptors. Each of the streams is decodable in a stand-alone fashion and the more descriptors we are getting, the higher is the video quality.

The major drawback of MDC is generation of the overhead by source splitting process. The overhead can be defined as the amount of data of the splitted streams in comparison with a single stream [6]. The overhead is generated by the encoding process and the IP network. In order to reduce the total overhead and to make MDC attractive even for bandwidth-limited links, it is desirable to apply header compression schemes. Since different descriptors can be sent over different channels, MDC inherently supports multi channel communication and it is particularly suitable for combination with cooperative header compression. For our investigation we assume that the number of channels is the same as the number of descriptors.

In Figures 7 and 8 the encoder and network overhead for the `foreman` and `container` video sequences using MDC is given. One can observe that the overhead depends on the video content, as well as the number of descriptors. The quantization value of 41 is chosen, since it guarantees high video quality and at the same time supports low data rates: the encoding overhead with quantization value 41 is not very high (see Figures 7 and 8, line ’QP 41 zero’). In case when large quantization values are used (like QP 41), the produced packets are rather small and the overhead increases drastically due to the network: if we are using RTP/UDP/IP suite for transportation of data through a network, an extra overhead of 40 bytes per packet for each descriptor should be taken into account if IPv4 is used (and it is 60 bytes for IPv6). The line ’QP 41 uncomp’ shows the total overhead in case no header compression is applied.

Applying header compression techniques, we can reduce the total overhead significantly. In Figures 7 and 8 the lines ’QP 41 AIC full’ corresponds to the case when each of the parallel channels will carry AICs for each of the other channels, therefore, if, for example, 8 descriptors are used, then 7 AICs are appended to each compressed header. This scheme will be very robust against channel errors, and from the figures we observe the reduction in the overhead. To reduce the overhead even further, we consider a cooperative scheme when each of the channels are carrying 3 AICs. One can imagine that the channels are grouped in clusters of 4 channels that ”help” each other: channel No. 1 carries additional information for channels No. 2, 3 and 4; channel No. 2 helps to the channels No. 3, 4, and 5 etc.

Using formulas from Section IV.A, one can find out that the number of cooperative channels equal to 4 is enough to maintain low PEP. At the same time, the total overhead is reduced from 200% to approx. 100% (see line ’QP 41 AIC 3’). We can conclude that cooperative header compression scheme with three AICs is a robust and bandwidth efficient scheme.

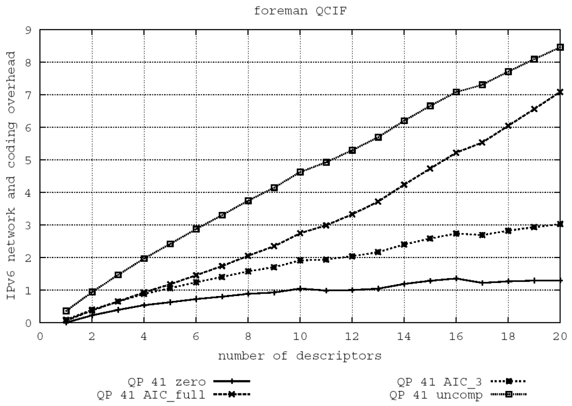

Fig. 7. Network overhead (RTP/UDP/IPv6) for the foreman video sequence and the quantization parameter 41.

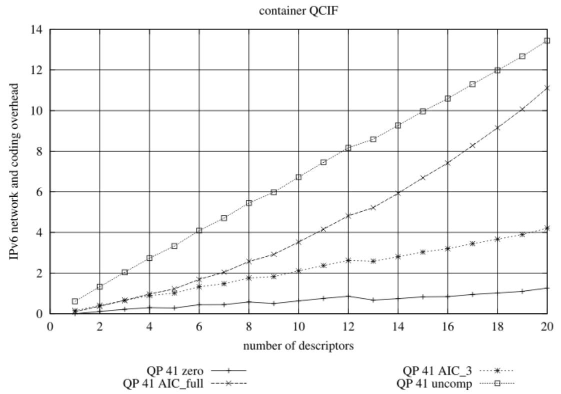

Fig. 8. Network overhead (RTP/UDP/IPv6) for the container video sequence and the quantization parameter 41.

V. CONCLUSIONS

This paper presents performance evaluation of header compression in case of correlated multiple channels. We propose an approach to model the correlation and investigate the impact of the error correlation on the packet error probability and bandwidth efficiency. Furthermore, we show that a scheme without feedback can be more efficient in terms of bandwidth usage compared with the compression relying on the receiver feedback for context updates. With the example of MDC, reduction in network overhead is illustrated.

REFERENCES

[1] F.H.P. Fitzek, S. Hendrata, P. Seeling, M. Reisslein. Chapter in Wireless Internet – Header Compression Schemes for Wireless Internet Access. CRC Press. 2004.

[2] S. Casner and V. Jacobson, Compressing IP/UDP/RTP Headers for Low-Speed Serial Links, Request for Comments 2508, February 1999.

[3] F.H.P. Fitzek, T. K. Madsen, P. Popovski, R. Prasad and M. Katz. Cooperative IP Header Compression for Parallel Channels in Wireless Meshed Networks. In *IEEE International Conference on Communication (ICC)*, Seoul, Korea, 2005.

[4] T. Madsen, F. Fitzek, Y. Takatori, R. Prasad and M. Katz, ”Cooperative IP Header Compression using Multiple Access Points in 4G Wireless Networks”, in *Proceedings of IST Wireless Summit*, 2005.

[5] J.G. Apostolopoulos, Error Resilient Video Compression Through The Use of Multiple States, in *IEEE Int. Conf. on Image Processing*, Vancouver, Canada, 2000.

[6] F.H.P. Fitzek, B. Can, R. Prasad and M. Katz. Overhead and Quality Measurements for Multiple Description Coding for Video Services. In *Proceedings ofWireless Personal Multimedia Communications (WPMC)* Padova, Italy, 2004.