Evaluation of a Generic Unidirectional Header Compression Protocol

Alexandre Soro* Jérôme Lacan† Emmanuel Chaput‡
Christophe Donny§ Cédric Baudoin¶

*TéSA and ENSICA, Univ. of Toulouse, 1, place E. Blouin, 31056 Toulouse Cedex, France
†ENSICA/TéSA/LAAS, Univ. of Toulouse, 1, place E. Blouin, 31056 Toulouse Cedex, France
‡INPT/IRIT/TéSA, Univ. of Toulouse, 2, rue Camichel - BP 7122 31071 Toulouse, France
§CNES, 18, avenue E. Belin, 31401 Toulouse, Cedex 4, France
¶Thales Alenia Space, 26, avenue J.F. Champollion, B.P. 1187, 31037 Toulouse Cedex 1 - France
Email: alexandre.soro@ensica.fr, jerome.lacan@ensica.fr, emmanuel.chaput@enseeiht.fr,
christophe.donny@cnes.fr, cedric.baudoin@thalesaleniaspace.com

***Abstract*—Header Compression techniques are now widely used in wireless and satellite communications. The main drawback of these techniques is to weaken the transmission against bit error or packet losses. Indeed, a corrupted or missing header can lead to a non-decompression of consecutive packets and then to a disconnection until the reception of a non-compressed packet. The parameters of the header compression system should then be carefully determined. In this paper, we first review the main header compression protocols standardized for a unidirectional link. This analysis allows us to build a simple generic header compression model depending on few parameters characterizing a header compression protocol. The evaluation of this model in cases corresponding to particular applications allows us to draw some first lessons for the use of header compression in Satellite communications.**

***Index Terms*—Header compression protocols, packet error rate, reliability, resynchronization time**

I. INTRODUCTION

The convergence of technologies has generalized the use of IP protocols in most network communications. Even if this generalization allows the various technologies to communicate, it implies the addition of new protocols in the protocol stack leading to more and more headers. For applications using small or medium packets (*e.g* voice over IP), the headers can represent an large part of the data. In wireless (including satellite) communications, the constraints in terms of bandwidth and loss recovery delay can largely benefit from header compression (HC) techniques which allow to reduce the size of the headers.

The main drawback of such techniques is to weaken the transmission against bit error or packet losses. Indeed, the loss of some packets can then lead to a non-decompression of the following packets headers and then to the loss of the corresponding payloads. Since wireless or satellite communications are subject to errors or losses, the HC protocols must be carefully designed.

Satellite communications can strongly benefit from HC, as any other communication means. However, some properties must be considered in the setting of the parameters of the HC protocols. First, the extremely large Round-Trip Time (RTT), roughly 500 ms for geostationary satellite (merely one second in a DVB-S/DVB-RCS scenario), can have a strong impact on the protocols using bidirectional links. Moreover, a large proportion of satellite applications does not have return link (*e. g.* DVB-SH [1]) and then can not safely use bidirectional compression protocols. The second property is that, contrary to some 3G-based protocol stacks, protocol stacks used in satellite communications (MPE, ULE, AAL5 and now GSE) do not allow the error bit to pass up to the link layer. Thus, the channel observed by the HC protocol is a packet erasure channel.

In order to evaluate and parametrize HC techniques on satellite communications, we need a model integrating the satellite properties. This paper proposes a first step toward this model by defining a generic model for an unidirectional link. After presenting the context in Section II,we present our model in Section III and discuss the results in Section IV.

II. UNIDIRECTIONAL HEADER COMPRESSION PROTOCOLS AND SATELLITE COMMUNICATIONS

Since unidirectional links can only use connection-less protocols, we only focus here on HC protocols for RTP/UDP/IP. Two main standardized header protocols can be used for this protocol stack: ROHC [2] and eCRTP [3].

eCRTP [3] is an enhanced version of CRTP [4] for links with high delay, packet loss and reordering. The robustness is mainly obtained by sending $N+1$ consecutive uncompressed packets after each change in a full value or a delta value, where N is a parameter representing the quality of the link between the hosts. In case of losses, the receiver tries to recover the header with the TWICE [5] algorithm. On unidirectional links, periodical refreshes are used.

Thanks to the use of the W-LSB compression method, ROHC (RObust Header Compression) [2] is probably the most

---

This work was done jointly by TéSA, Thalès Alenia Space and CNES, and was funded by CNES.

efficient HC protocol. Three compression states are defined for the compressor and the decompressor. Orthogonally to the states, the ROHC scheme has three modes of operation, called Unidirectional (U), Bidirectional Optimistic (O), and Bidirectional Reliable (R) mode. In the unidirectional mode, the transition between the compression states, and thus the refreshes of the static and dynamic contexts are determined by time-out parameters. In the bidirectional Optimistic mode, transitions to a higher level of compression are obtained by the similar optimistic approach used in U mode. However, transitions back to a lower compression level are linked to reception of negative acknowledgements from the decompressor. Finally, the bidirectional reliable mode completely rests on acknowledgements sent by the decompressor. Positive acknowledgements (ACK) make the compressor transit to a higher compression state, and negative acknowledgements (NACK), similarly as O mode, get the compressor to a lower compression level. These characteristics seem to show that this last mode is less appropriate in a satellite context, for example.

Evaluations and comparisons of these protocols were proposed in *e. g.* [6] or [7], however, these papers does not integrate the specificities of the satellite context, as it does not allow bit errors to pass up the link layer. The first step toward a model considering this parameter is presented in the next Section as a comparison model.

It should be noted that a direct application target for this model could be ROHCv2 [8], as drafts uses a similar 2-state compression model. Indeed, first works on ROHCv2 use 2 states: one corresponding to IR-state as ROHCv1, and the other one corresponding to a general compression state. It is worth noticing that in the compression state (CO), the compression is dynamic. Given the previous packets sent, the compressor supposes the decompressor state and encodes the packet with the appropriate method. Moreover, the compressor must consider the optimistic approach which is quite similar to the ROHCv1 approach.

III. MODELING AND ANALYSIS OF A GENERIC MODEL OF UNIDIRECTIONAL HEADER COMPRESSION PROTOCOL

A. *Introduction of the model*

This paper aims to evaluate the influence of the different parameters of the system on a HC protocol behavior, and more particularly on the packet loss rate. Indeed, a loss of a packet with a compressed header on the channel can lead to non-decompression of the following packets. These packets can not be used by the upper layers, even if they were correctly received. It follows that the Packet Error Rate (PER) at the output of the HC layer is necessarily greater than the Frame Error Rate (FER) at the input of the HC layer. Fig. 3 illustrates this process. The main achievement of our model is to estimate the PER from the FER and the different HC parameters. For that, we define a simple generic HC model ecompassing the main concepts used by [2] and [3] on an unidirectional link.

The model is a 2-state compression model. Transitions between these states are periodical. In the first state, n consecutive packets with uncompressed headers are sent. This type of header will be used for context recovery since every packet of one burst of uncompressed packets carries the context. Indeed, reception of any packet of this burst will recover the context. The average length of these headers is represented by l_n. The second state corresponds to a burst of c consecutive packets with compressed headers. Their average header length is l_c. Representation of the state machine and its chronological evolution are given in Fig. 1 and Fig. 2.

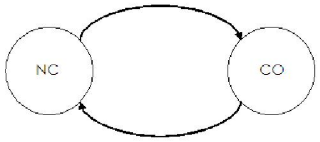

Fig. 1. State machine of the compressor

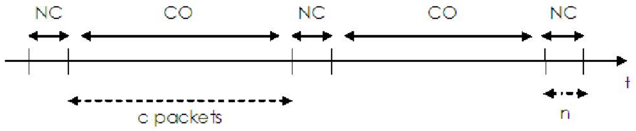

Fig. 2. Chronological evolution of the compressor

We consider a packet erasure channel (*i.e.* the packets are either lost or received without errors) with independent losses.

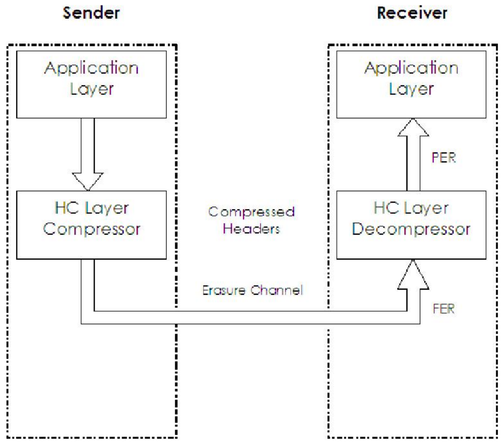

Fig. 3. Transmission scheme

Our model considers that when the context is lost, there are two ways to recover it :

1) with a successful reception and a validation of a compressed header of the same burst (*i.e* all consecutive packets of the same type).

2) with a successful reception of an uncompressed header.

Indeed, when a compressed header packet is lost, the context could be recovered by the successful validation of the first received packet from the remaining ones of the burst (case 1). If a compressed header of the same burst is successfully received but does not allow to recover the context, we are in an invalidation state. Then the context could be only recovered by uncompressed headers (case 2).

When an uncompressed header packet is lost, it could be recovered by remaining packets of the same burst, compressed headers of the following CO burst or any other uncompressed header. Same rules applies to validation of compressed headers.

Validation consists in the fact that compressed headers do not carry the entire information of the header. This compression is lossy. So, when the context is lost, and the first following packet to be received is a compressed header one, recovering the context is not guaranteed. This probability depends on the gap between the loss of synchronization consecutive to a loss, and the first compressed header received. The number of packets in the gap is denoted by δ. For any δ, a corresponding probability of successful recovery is applied: $p(\delta)$. This probability only applies to compressed headers as uncompressed headers always help to recover the context.

This parameter, which has a strong influence on the performance, depends totally on the recovery mechanisms used: TWICE for eCRTP, or W-LSB for ROHC, for example.

B. Analytical study

The evaluation of the PER from the FER and the HC parameters is done in two steps. In the first step, we consider the following event at the receiver side : a frame is lost given that the previous frame was received and the context was successfully validated. In this case, our objective is to evaluate the average number of packets lost in output. This average number is denoted by μ. The second step consists in estimating the occurrence probability of this event.

The value of the parameter μ depends if the lost frame has a compressed or an uncompressed header. Hence, we will define respectively the parameters μ_c for compressed headers and μ_n for uncompressed headers.

We number the packets of the same compressed burst as shown in Fig. 4.

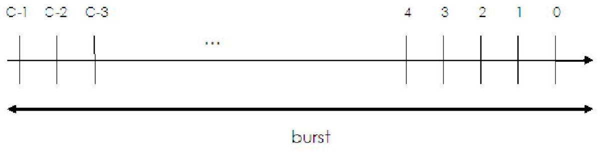

Fig. 4. A compressed burst

Same notation is used for uncompressed headers. For i varying from 0 to $C-1$, we denote by $\mu_C(i)$ the average number of lost packets following the loss of the compressed header i. We have :

$$\mu_c = \frac{1}{c} \times (\mu_c(0) + \mu_c(1) + ... + \mu_c(I-1))$$

We introduce the average number of packets lost when the context is recovered by an uncompressed header excluding the number of packets remaining in the lost burst: NR_C for "not recovered by a compressed header". We have :

$$\begin{aligned} NR_C &= 1 + z_n(1 - FER^n) \\ &\quad +(z_n + (n+c))(1 - FER^n)FER^n \\ &\quad +(z_n + 2(n+c))(1 - FER^n)FER^{n\times 2} \\ &\quad +... + (z_n + i(n+c))(1 - FER^n)FER^{n\times i} \end{aligned}$$

Then,

$$NR_C = 1 + z_n + (n+c) \times \frac{FER^n}{1 - FER^n}$$

where z_n represents the fact that it may not be the first packet of an uncompressed burst that can recover the context. We have:

$$\begin{aligned} z_n &= \frac{1}{1 - FER^n} \times \Big(0 \times (1 - FER) \\ &\quad +1 \times (1 - FER)FER \\ &\quad +2 \times (1 - FER)FER^2 \\ &\quad +... + (n-1) \times (1 - FER)FER^{n-1}\Big) \\ &= \frac{1}{1 - FER^n} \times \Big(\sum_{i=1}^{n-1} i \times FER^i \; (1 - FER)\Big) \end{aligned}$$

Finally,

$$z_n = \frac{FER}{1 - FER} - n \times \frac{FER^n}{1 - FER^n}.$$

Then, for $i > 0$,

$$\begin{aligned} \mu_c(i) &= 1 \times (1 - FER) \; P(1) \\ &\quad +2 \times (1 - FER) \; FER \; P(2) \\ &\quad +... + \; i \times (1 - FER) \; FER^{i-1} \; P(i) \\ &\quad + \; (i + NR_C)(1 - X_i) \\ &= (1 - FER) \; \sum_{k=1}^{i} k \; FER^{k-1} \; P(k) \\ &\quad +(i + NR_C)(1 - X_i) \end{aligned}$$

with $X_i = (1 - FER) \sum_{k=1}^{i} FER^{k-1} \; P(k)$. Finally :

$$
\begin{aligned}
\mu_c \;=\; & \frac{1}{c} \times \Bigg(NR_C \\
& + \sum_{i=1}^{C-1} \Big((1 - FER) \sum_{k=1}^{i} k \; FER^{k-1} \; P(k) \\
& + (i + NR_C)(1 - X_i) \Big) \Bigg)
\end{aligned}
$$

For the resolution of μ_n, we apply same techniques, which give:

$$
\mu_n(0) = (1 - FER) \sum_{k=1}^{C} k \, P(k) \; FER^{k-1} + (c + NR_C)(1 - Y_0)
$$

and for $i > 0$:

$$
\begin{aligned}
\mu_n(i) \;=\; & (1 - FER^i) \\
& + (1 - FER) \sum_{k=1+i}^{C+i} k \; P(k) \; FER^{k-1} \\
& + (c + i + NR_C)(1 - Y_i)
\end{aligned}
$$

with $Y_i = 1 - FER^i + (1 - FER) \sum_{1+i}^{1+C} FER^{k-1} \; P(k)$.

Finally, we have:

$$
\begin{aligned}
\mu_n \;=\; & 1 + \frac{1}{n} \Big(- \frac{1 - FER^n}{1 - FER} \\
& + (1 - FER) \sum_{i=0}^{n-1} \sum_{k=i+1}^{i+c} k \; FER^{k-1} \; P(k) \\
& + (c + NR_C) \sum_{i=0}^{n-1} (1 - Y_i) + \sum_{i=0}^{n-1} i \; (1 - Y_i) \Big)
\end{aligned}
$$

In order to determine the output error rate, we have to consider that when a loss occurs, it may not have consequences on the output error rate. This event corresponds to the fact that the loss is included in the consequences of a previous loss. This behavior is illustrated in Fig. 5.

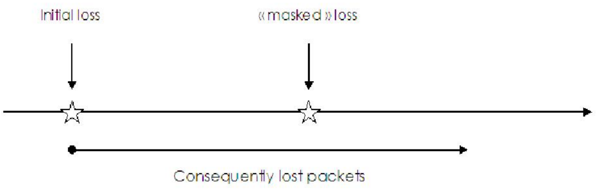

Fig. 5. Illustration of a masked frame loss

Therefore, we have to determine the probability of a loss not being "masked". We notice that this event is strictly equivalent to the context being synchronized before this loss. Since this probability depends wether the header is compressed or not, we introduce the two following probabilities: $P(Context\ OK/NC)$ and $P(Context\ OK/CO)$. These probabilities have not to be confused with $P(\delta)$. Then, we have:

$$
\begin{aligned}
PER \;=\; & \frac{FER}{n + c} \times \Big(n \; \mu_N \times P(Ctxt\ OK/NC) \\
& + c \; \mu_C \times P(Ctxt\ OK/CO) \Big)
\end{aligned}
$$

To determine $P(Context\ OK/NC)$ and $P(Context\ OK/CO)$, we take the approximation that in steady state, for an output packet error rate PER, the probability that the context is synchronized is $1 - PER$. However, we can improve this approximation by considering that when the previous packet before the loss is an uncompressed one, the properties of these packets give us that the context is synchronized if and only if this packet is received. In this case, the probability that the context is synchronized is $1 - FER$. Thus, we have:

$$
P(Context\ OK/NC) = 1 - \frac{(n - 1)FER + PER}{n}
$$

$$
P(Context\ OK/CO) = 1 - \frac{FER + (c - 1)PER}{c}
$$

Finally:

$$
PER = FER \times \frac{n \; \mu_n + c \; \mu_c - FER\,((n - 1)\mu_n + \mu_c)}{n + c + FER\,(\mu_n + (c - 1)\mu_c)}
$$

We also introduce the efficiency parameter e which corresponds to the ratio between the average header size and the size of uncompressed headers :

$$
e = \frac{n \; l_n + c \; l_c}{(n + c) \; l_n}
$$

where l_n and l_c are respectively the size of uncompressed and compressed headers.

Finally, we define the corresponding resynchronization time between the compressor and the decompressor, *i. e.* when the decompressor can not decompress the headers following a packet loss. This time, which gives an indication of a consequence of a lost packet, is defined as following:

$$
t = \mu \times \frac{l_p + l_n.e}{R},
$$

where R is the constant transmission rate and l_p is the average payload size, considering an overall average μ as $\mu = \frac{n \; \mu_n + c\mu_c}{n+c}$.

C. Results

We give examples or results obtained with this model. Figure 6 shows the influence of c, the number of consecutive compressed headers, and the recovery performance of the protocol $p(\delta)$ on the value of μ. Note that $p(\delta)$ was modeled by the function y^{δ}, where y varies between 0 and 1. The others parameters are fixed as follows : $n = 3$, $FER = 10^{-4}$, $l_n = 40$ bytes, $l_c = 4$ bytes, $l_p = 450$ bytes and $R = 100$ KBytes/s.

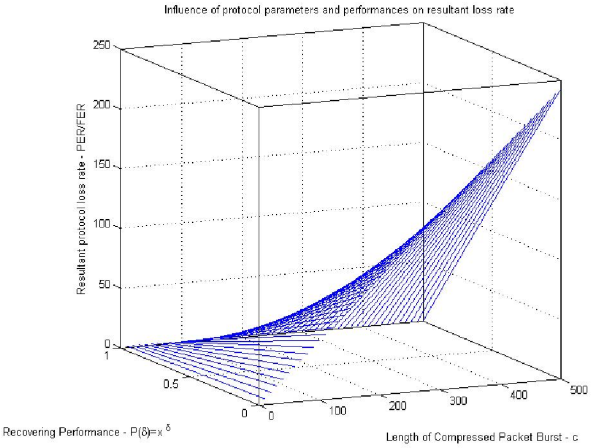

Fig. 6. Influence of protocol parameters and recovery mechanisms performance on the output packet error rate

Figure 7 shows the values of the resynchronization time in function of the input frame loss rate (FER) and the transmission rate (in bytes). The others parameters are fixed as follows : $n = 3$, $c = 50$, $l_n = 40$ bytes, $l_c = 4$ bytes, $l_p = 450$ bytes and $p(\delta) = 0.7^{\delta}$.

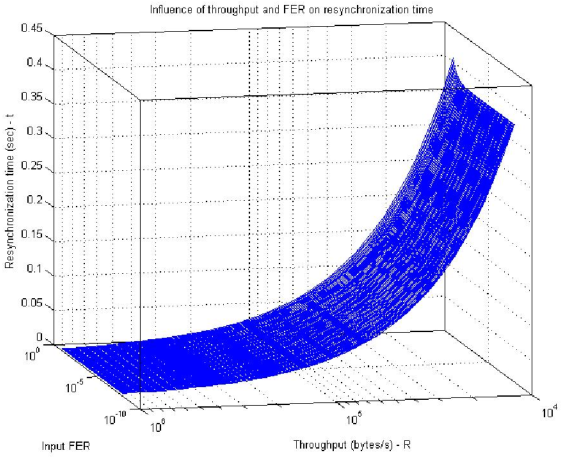

Fig. 7. Influence of FER and transmission rate on the resynchronization time.

IV. DISCUSSION AND CONCLUSION

Some first lessons can be drawn from Figures 6 and 7. Indeed, Figure 6 shows that, for classical parameters, the multiplicative factor between the input PER and the output PER can reach two orders of magnitude and then, can directly cause the mis-functioning to some applications (e.g. video).

Figure 7 shows that the resynchronization time is, for classical parameters, less than 0.15 seconds. This is a very interesting information in the satellite context because, for a bidirectional HC protocol using context refreshes based feedback of the decompressor (*e.g.* modes R and O of ROHC), the resynchronization time is equal to the round trip time (RTT). Thus, the implication of the Figure 7 is that, for the consider parameters, a bidirectional mode is useless for satellite communications.

This work will be extended in several ways. First, the proposed two-states model will be extended to a three-states model to evaluate unidirectional ROHC. The integration of the return link, and thus of the RTT parameter, is also planned, in order to compare the different modes of ROHC. An accurate analysis of the performance of the recovery mechanisms like W-LSB or TWICE, will also be performed. The obtained model will be then evaluated on erasure channels integrating burst losses patterns.

REFERENCES

[1] "Digital Video Broadcasting (DVB); Framing Structure, channel coding and modulation for Satellite Services to Handheld devices (SH) below 3 GHz," 2007.

[2] C. Bormann et al., "RObust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed ," RFC 3095 (Proposed Standard), IETF, 2001.

[3] T. Koren, S. Casner, J. Geevarghese, B. Thompson, and P. Ruddy, "Enhanced Compressed RTP (CRTP) for Links with High Delay, Packet Loss and Reordering," RFC 3545 (Proposed Standard), IETF, 2003.

[4] S. Casner and V. Jacobson, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links," RFC 2508 (Proposed Standard), IETF, 1999.

[5] M. Degermark, B. Nordgren, and S. Pink, "IP Header Compression," RFC 2507 (Proposed Standard), Feb. 1999. [Online]. Available: http://www.ietf.org/rfc/rfc2507.txt

[6] H. Jin, R. Hsu, , and J. Wang, "Performance comparison of header compression schemes for rtp/udp/ip packets," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2004.

[7] A. Couvreur, L.-M. Le-Ny, A. Minaburo, G. Rubino, B. Sericola, and L. Toutain, "Performance analysis of a header compression protocol: The rohc unidirectional mode," *Telecommunication Systems*, vol. 31, pp. 85–98, 2006.

[8] G. Pelletier and K. Sandlund, "RObust Header Compression Version 2 (RoHCv2): Profiles for RTP, UDP, IP, ESP and UDP Lite," draft-ietf-rohc-rfc3095bis-rohcv2-profiles-01, Internet Draft, work in progress, 2007.