Optimizing the Error Recovery Capabilities of LDPC-staircase Codes Featuring a Gaussian Elimination Decoding Scheme

Mathieu CUNCHE   Vincent ROCA

INRIA Rhône-Alpes, Planète research team, France, {firstname.name}@inria.fr

Abstract— **This work focuses on the LDPC codes for the packet erasure channel, also called AL-FEC (Application-Level Forward Error Correction codes). Previous work has shown that the erasure recovery capabilities of LDPC-triangle and LDPC-staircase AL-FEC codes can be greatly improved by means of a Gaussian Elimination (GE) decoding scheme, possibly coupled to a preliminary Zyablov Iterative Decoding (ID) scheme. Thanks to the GE decoding, the LDPC-triangle codes were very close to an ideal code. If the LDPC-staircase performances were also improved, they were not as close to an ideal code as the LDPC-triangle codes were.**

The first goal of this work is to reduce the gap between the LDPC-staircase codes and the theoretical limit. We show that a simple modification of the parity check matrix can significantly improve their recovery capabilities when using a GE decoding. Unfortunately the performances of the same codes featuring an ID are negatively impacted, as well as the decoding complexity. The second goal of this work is therefore to find an appropriate balance between all these aspects.

Index Terms— **AL-FEC codes, erasure channel, LDPC-staircase, hybrid iterative decoding/Gaussian elimination**

I. INTRODUCTION

This work focuses on the design of LDPC large block FEC codes for the packet erasure channel (or similarly the Bit Erasure Channel, BEC). Since these codes are often used within the application (or transport) layer, they are usually called AL-FEC.

AL-FEC codes are commonly implemented as software codecs. This is a key advantage that provides a lot of flexibility. Indeed, the block size (i.e., the number k of source symbols in the source block that is FEC encoded) and the code rate can be easily tailored to the application requirements, just on time. Besides these codes usually take advantage of the host memory, a plentiful resource, whereas hardware codecs (e.g., implementing physical layer FEC codes) are limited by the available chipset memory. This feature enables in particular LDPC codes to work efficiently on very large objects and encoding blocks.

FEC codes for the erasure channel are therefore the cornerstone of many content delivery protocols and systems. More precisely, AL-FEC codes are a key building block of the FLUTE/ALC protocol stack designed by the Reliable Multicast Transport (RMT) working group of the IETF. The LDPC-staircase codes considered in this work have recently been standardized by the RMT working group in RFC 5170 [11] as one AL-FEC scheme among others (e.g., Reed-Solomon

and Raptor codes). Since FLUTE/ALC are integrated in the DVB IP Datacasting service for reliable file delivery, AL-FEC codes are naturally a key building block for DVB systems. Additionally, AL-FEC codes can also be used at the link layer, as in in the MPE-FEC layer of DVB-H systems or the MPE-IFEC layer of DVB-SH systems. In that case they are known as the Upper Layer FEC codes, or UL-FEC (here "upper" must be understood as being above the physical layer). More generally, all the transmission systems that behave as an erasure channel can take advantage of AL-FEC codes.

We have shown in [5] that the erasure recovery capabilities of the LDPC-triangle and LDPC-staircase codes [10][11] can be greatly improved by means of a hybrid Zyablov Iterative Decoding (ID) [14]/Gaussian Elimination (GE) scheme. Thanks to this hybrid decoding scheme, the erasure recovery capabilities of LDPC-triangle codes were very close to that of an ideal code. However, there was still a gap between the performances of the LDPC-staircase and that of ideal codes.

A first goal of this work is to further improve the performances of LDPC-staircase codes. More precisely, this work studies the influence of the source node degree on the erasure recovery capabilities of the LDPC-staircase codes featuring a hybrid decoding scheme. We show that the performances can easily approach the theoretical limits. Unfortunately the performances of these codes featuring an ID are negatively impacted, as well as the decoding complexity. A second goal of this work is therefore to find an appropriate balance between all these aspects and to provide practical recommendations on how to use these codes.

The remaining of this paper is organized as follows : Section II introduces some related works. Then section III introduces the LDPC codes considered and the hybrid decoding scheme used. Section IV gives an account of performance measurements. Finally we give practical recommendations on the use of these codes and we conclude.

II. RELATED WORKS

As explained above, we comprehensively studied the use of a hybrid ID/GE scheme with LDPC-staircase and triangle codes in [5]. Independently and in parallel, the authors of [7][8] did the same with different LDPC codes and, non surprisingly, came to the same conclusions.

Improved approaches for GE decoding of LDPC codes are presented in the following patent description [12] and in [3].

Thanks to a procedure called symbol deactivation in [12] and reference variables declaration in [3], the GE is applied on a reduce linear system. The remaining variable are then deduced by substitution.

In [4] a scheme called In-place algorithm is presented, which uses a Gaussian elimination for decoding LDPC codes and other binary linear block codes on the BEC.

In [2] and [1] the authors introduce a Generalized Iterative Decoding method for LDPC codes. At each iteration a linear system of size p (the order of the decoder) is solved. This method reduces the complexity since the decoder solves several smaller linear systems instead of a big one. This method can be seen as an hybrid decoder : decoding starts with a decoder of order 1 (which is a standard ID) and the order of the decoder is increased until the decoding succeeds.

In our work, we decided to focus on two well known, non patented decoding techniques for the erasure channel. However, additional work is still needed to further improve the decoding performances and some of the above references might be useful to that purpose. This is left to future works.

III. THE LDPC-STAIRCASE CODES AND THE HYBRID ID/GE DECODING SCHEME

We start by describing the LDPC codes considered, as well as the hybrid ID/GE scheme.

A. The LDPC-staircase Codes

The LDPC-staircase codes [6][11] are very simple LDPC codes that, by design, feature a very fast encoding. Their parity check matrix H, of size $n - k$ rows for n columns, is built as follows : H is the concatenation of a left sub-matrix H_1, of size $n - k$ rows for k columns (i.e., the source symbols), and a right sub-matrix H_2, of size $n - k$ rows for $n - k$ columns (i.e., the parity symbols).

With the codes considered, the H_2 sub-matrix has a "staircase" structure (also called double diagonal) structure. It means that each parity symbol is the XOR sum of the previous parity symbol plus a very small number of source symbols. The H_1 sub-matrix is filled in a fully regular way, thanks to the following algorithm :
- *step 1 :* insert N1 1-s randomly but evenly into each column ;
- *step 2 :* check that there are at least two 1-s per row. If not, add one or two 1-s randomly to these rows until the condition is satisfied ;

Because of this second step, the average number of 1-s per column in the H_1 matrix (i.e., the source node degree) can be greater than the specified N1 parameter with small code rates.

In [10] the N1 value was fixed and set to 3 because the ID algorithm shows its optimal performance with this value, while keeping the encoding/decoding complexity to a minimum. As we will see later on, this choice is no longer appropriate when a GE scheme is used. In RFC 5170, N1 is a parameter that is now communicated to the decoders and whose default value is 3. The interested reader is invited to refer to [11] for a detailed specification of LDPC-staircase codes.

B. The Hybrid ID/GE Scheme

The ID algorithm has a very low complexity but it is known to be suboptimal in terms of erasure recovery capabilities, especially with regular LDPC codes. On the opposite, the GE algorithm can solve any non-singular linear system at the price of a higher algorithmic complexity.

In [5] we proposed a hybrid decoding scheme that works as follows :
- start decoding with the ID algorithm ;
- then, if it fails and if it is known that no additional symbol will be received, switch to the GE algorithm ;

This hybrid approach :
- significantly reduces the average decoding complexity, since the complex GE is only used when needed, over a system of linear equations that has been simplified (up to a certain point that depends on the actual channel loss probability) by the ID ; and
- achieves the optimal erasure recovery capabilities made possible by the underlying FEC codes.

The choice of switching to GE can also depend on additional parameters. For instance, a receiver might decide that a GE will only be used if the simplified system (i.e., after ID) has a size inferior to a certain threshold. This threshold can take into account the processing capabilities of the terminal, or the remaining battery capacity, or the estimated decoding time. This flexibility is another key benefit of this hybrid decoding scheme.

IV. RESULTS

We now analyze the performances of the LDPC-staircase codes, both from an erasure recovery capability point of view and from a decoding complexity point of view. Our goal is to be able to provide guidelines for an optimal use of these codes.

A. Experimental Setup

These tests use the high performance, open-source, LDPC C++ codec, version 2.1 [13], for which we added a GE scheme in addition to the existing ID.

Once encoding has been performed, the source and repair symbols are transmitted in a fully random order. A loss probability is then applied, leading to the removal (i.e., erasure) of a certain number of symbols. The receiving application submits each received symbol to the decoder. This application stops either when the decoding finishes or when all symbols have been received and submitted to the decoder. The decoding status (success or failure) is then determined. Note that there is no explicit loss model (e.g., random or per burst erasures) because shuffled transmissions are not affected by this model. The loss probability (for a given code rate) is the only parameter that needs to be considered.

In the following experiments, we consider that the decoder uses :
- either a GE-only scheme : this is useful to appreciate the maximum erasure recovery capabilities of LDPC-staircase codes ;

- or a hybrid decoding scheme : this is useful to appreciate the decoding complexity, depending on the symbol loss probability ;
- or an ID-only scheme : this is useful to appreciate the impacts of a choice for N1 on decoders that do not feature (or use) any GE ;

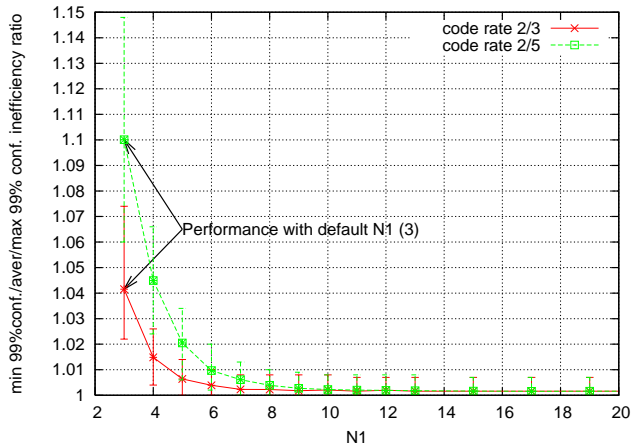### B. Erasure Recovery Capabilities with the GE Scheme

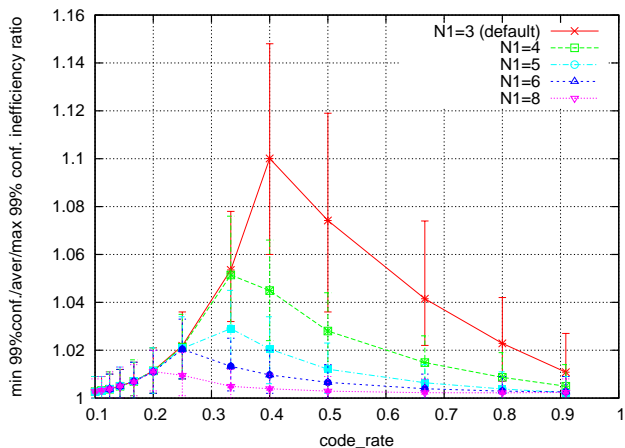Fig. 1. Inefficiency ratio w.r.t. the N1 parameter (object composed of 1,000 symbols).

Fig. 2. Inefficiency ratio w.r.t. the code rate (object composed of 1,000 symbols).

We start by giving an account of experiments meant to assess the erasure recovery performances when a receiver uses a GE-only decoding scheme or a hybrid decoding scheme (the initial use of an ID scheme does not impact the erasure recovery performances).

This erasure recovery capability can be evaluated by means of the average (plus the 99% percentiles above or below this average) "inefficiency ratio", which is the ratio of the number of symbols needed for decoding to complete successfully over the number of source symbols. We sometimes consider the "overhead", i.e., the inefficiency ratio minus 1, expressed in percent.

The erasure recovery capability can also be evaluated by means of the decoding failure probability as a function of the loss probability. This is useful to evaluate the error floor.

For the experiments that evaluate the inefficiency ratio as a function of either the N1 value, or the code rate, or the object size, 10^3 tests are performed for each set of parameters. For the experiments that evaluate the decoding failure probability as a function of the loss probability, 10^6 tests are performed for each set of parameters in order to have the required precision.

Fig. 1 and Fig. 2 show that the inefficiency ratio decreases very quickly as N1 increases. There is therefore an incentive to increase the N1 value. However, it can be noticed that all the curves converge as the code rate approaches zero, because the N1 parameter that is used in the first step of the parity check matrix creation algorithm is no longer sufficient to guaranty that all equations contain at least two 1-s. As a consequence, the second step of the parity check matrix creation algorithm, which does not depend on N1, becomes predominant (Section III-A).
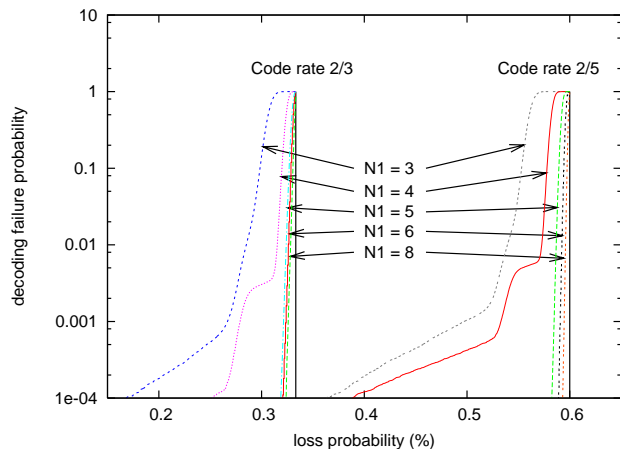

Fig. 3. Decoding failure probability w.r.t. the erasure probability (object composed of 1,000 symbols).

In Fig. 3 we see that increasing N1 improves the performance of the code in the waterfall region as well as in the error floor region. For instance (see Table I), with code rate 2/3 (resp. 2/5), using N1=5 makes the error floor lower than 10^{-4} for a 2.21% overhead (resp. 4.41%), while the average overhead amounts to 0.63% only (resp. 2.04%), which is a key achievement.

code rate	average overhead	overhead for a failure prob. $\leq 10^{-4}$	loss rate for a failure prob. $\leq 10^{-4}$
2/3	0.63%	2.21%	31.9%
2/5	2.04%	4.41%	58.2%

TABLE I

AVERAGE OVERHEAD AND MAXIMUM OVERHEAD FOR A FAILURE PROBABILITY LOWER THAN $10{-4}$ AS A FUNCTION OF THE CODE RATE (OBJECT COMPOSED OF 1,000 SYMBOLS, N1=5).

Finally, Fig. 4 shows, when N1=5, the performances of the LDPC-staircase codes as a function of the object size. If, non surprisingly, these codes perform well for big objects, we also
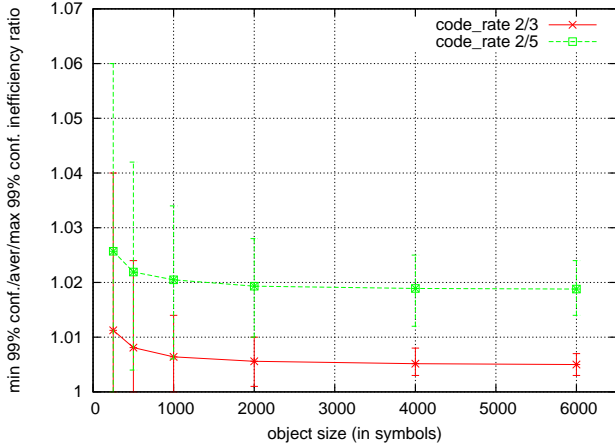
Fig. 4. Inefficiency ratio w.r.t. the object size (N1=5).

see that the same is true for small objects (of size 250 and 500 source symbols) where the performance penalty remains relatively limited.

C. Decoding complexity

We now give an account of experiments meant to assess the decoding complexity for receivers that feature a hybrid ID/GE scheme. In these tests, the object is composed of 1,000 source symbols, each symbol being 1024 byte long, and the code rate is set to 2/3. These experiments are performed on a Linux PC using a 2.6.18-6/64 bit Debian operating system and featuring a Dual Core Intel Xeon 5120/1.86GHz (1066MHz) processor. For each point 200 tests are performed and we plot the minimum/average/maximum decoding times (or the corresponding bit rate) *when decoding succeeds.*

For comparison purposes, we also use the excellent Reed-Solomon open-source C codec designed by L. Rizzo for $GF(2^8)$ [9], which has been widely used in the community. Because of the intrinsic limitations of a Reed-Solomon codec working on $GF(2^8)$, the source object is partitioned into several blocks such that : $k \leq n \leq maxn = 255$, using the block partitioning algorithm standardized for FLUTE/ALC.

The results, in terms of decoding time (logarithmic scale) or decoding throughput (linear scale) are shown respectively in Fig. 5 and Fig. 6. Non surprisingly, the N1 parameter has major impacts on the decoding complexity. This is caused by several phenomenons :

– the decoding complexity of the ID depends on the number of variables in each equation, which determines the number of XOR and elementary matrix management operations that are needed during decoding. This is the reason why the curves for low loss probabilities (i.e., when an ID always succeeds) are not superposed ;

– the decoding complexity of the GE depends on the size of the system of linear equations. With higher loss probabilities, the ID turns out to be inefficient. Therefore the decoding complexity directly derives from the system complexity over which a GE is performed. In that case, by increasing N1, we increase the number of 1-s in the

matrix and therefore the number of operations needed by the GE algorithm to diagonalize the matrix ;

The decoding complexity also depends on the experienced loss rate :

– the *average* decoding complexity depends on the probability that a GE is needed. By increasing N1, we decrease the probability that the very fast ID be successful, and a GE is more often needed. This is visible in the curves : with low loss probabilities, the curve is almost flat, meaning that ID is always sufficient. Then there is a knee and the average decoding complexity increases progressively since the GE scheme is more and more often needed.

– there is relationship between the loss probability and the size of the system that needs to be solved with a GE. However, it also depends on the loss model, and in particular what symbols have been erased.

Nevertheless, even if the decoding complexity of the LDPC decoder increases with N1 and with the loss probability, our decoder is, on average, at least one order of magnitude faster than the reference Reed-Solomon codec :

– Reed-Solomon : 54 Mbps on average (reference) ; [1]

– LDPC-staircase, N1=5, low loss probability, where ID is always successful : 1.75 Gbps (32.4 times faster) on average ;

– LDPC-staircase, N1=5, high loss probability, where GE is absolutely needed : 550 Mbps (10.2 times faster) on average ;

Let us now assume that N1=5 (which is often a good choice, but not necessarily, see section V). Fig. 7 shows that in all situations, LDPC-staircase case are significantly faster than Reed-Solomon codes over $GF(2^8)$. Of course, the larger the object, the more costly the Gaussian elimination. We see in particular that even with objects of size 4,000 source symbols (with 1024 byte symbols), LDPC-staircase codes remain faster than Reed-Solomon and provide significantly better erasure recovery capabilities. If decoding complexity is an issue with such a large object, choosing a smaller N1 value remains possible.

D. Erasure Recovery Capabilities with the ID Scheme

Finally we give an account of experiments meant to assess the performance of decoders that are limited to ID (no matter the reason, e.g., because they don't have the needed processing capabilities or they need to enter power-save mode). These tests are carried out under the same conditions as in section IV-B.

Fig. 8, 9 and 10 should be compared to their GE counterparts, i.e., respectively Fig. 1, 2 and 3. As we already said, N1 has globally a negative impact on the performances of the ID scheme, and Fig. 8 and 9 both show that the inefficiency ratio quickly increases with N1. However, Fig. 10 also show that increasing N1 greatly improves the performances in the error floor region. In particular, using N1=5 seems to offer

---

[1]Since the transmission order is randomized, the average percentage of repair symbols received is constant, no matter the loss probability, and only depends on the code rate. The Reed-Solomon decoding time, which depends on the number of source symbols to recover, is therefore constant during these tests.
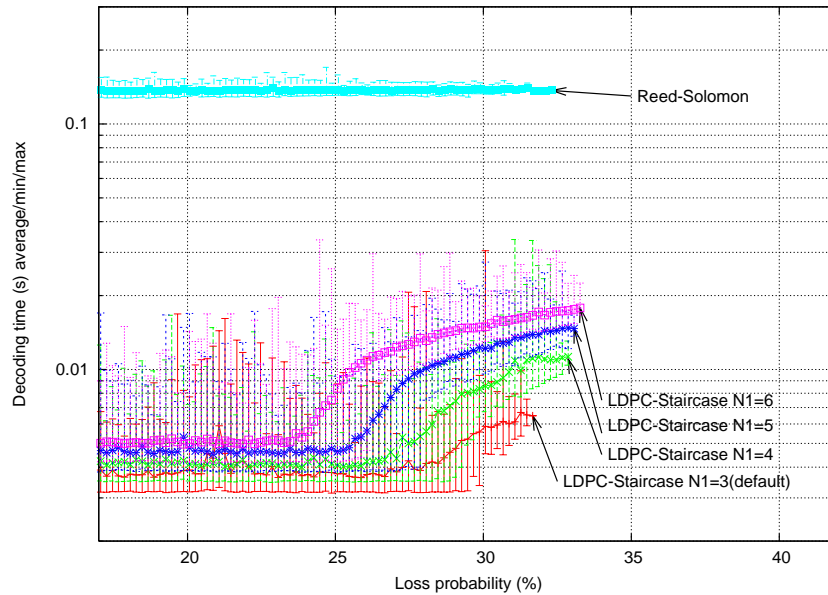
Fig. 5. Decoding time w.r.t. the loss probability (hybrid ID/GE scheme, object composed of 1,000 symbols, code rate 2/3).
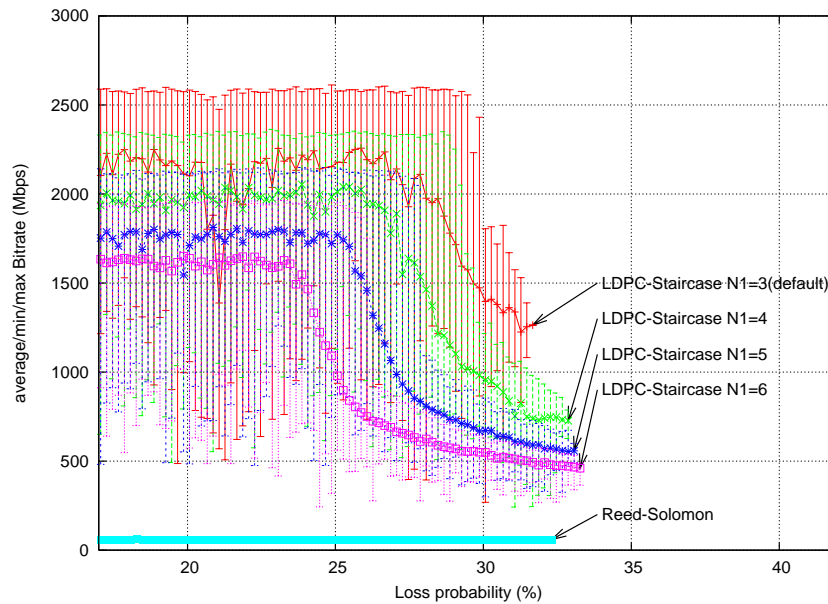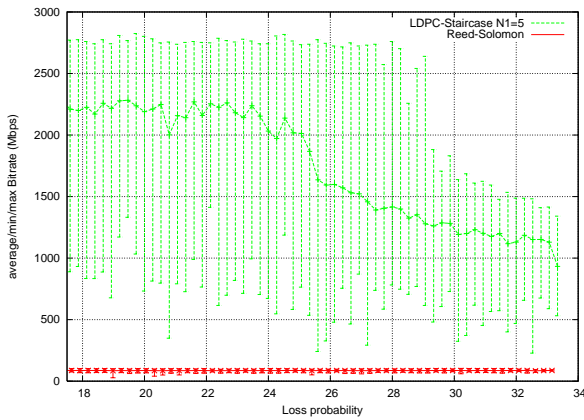

Fig. 6. Bitrate w.r.t. the loss probability (hybrid ID/GE scheme, object composed of 1,000 symbols, code rate 2/3).

a good balance. This is a good point that can compensate, up to a certain point, the overall erasure recovery capability degradation.

V. PRACTICAL RECOMMENDATIONS

Thanks to the previous tests, we can now draw some general guidelines on how to choose the N1 parameter :

– when it is known that *all* the receivers feature an ID-only scheme, using $N1 = 3$ (the default value) is a good solution, unless we want to favor a low error floor. In that case, using $N1 = 4$ or $N1 = 5$ is often a good alternative.

– when it is known that *some* receivers feature an ID-only scheme, the sender should use a N1 value that leads to good results with both decoders. $N1 = 4$ or $N1 = 5$ are often appropriate.

– when it is known that *all* receivers feature a hybrid decoding scheme, the sender should use a N1 value like N1=5 that leads to excellent erasure recovery capabilities, with a very small error floor, without impacting too seriously the decoding complexity.

– when the decoding complexity is the main issue, using the default N1=3 value remains acceptable with certain code rates. For instance, with a code rate equal to 0.8 or higher, few differences will be seen in practice in terms of erasure recovery capabilities, and using N1=3 keeps the decoding complexity to a minimum.

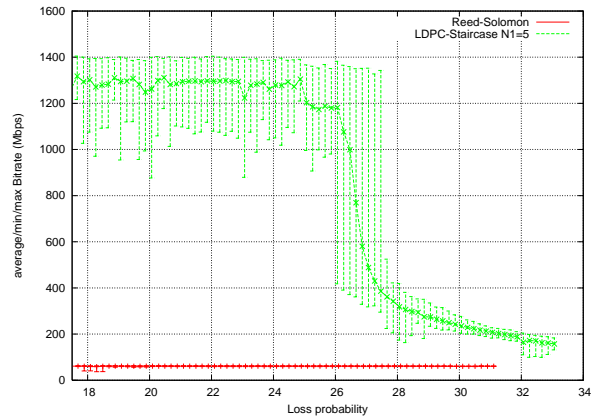– when the code rate is very small, equal to 0.25 or lower,

(a) object composed of 250 symbols.
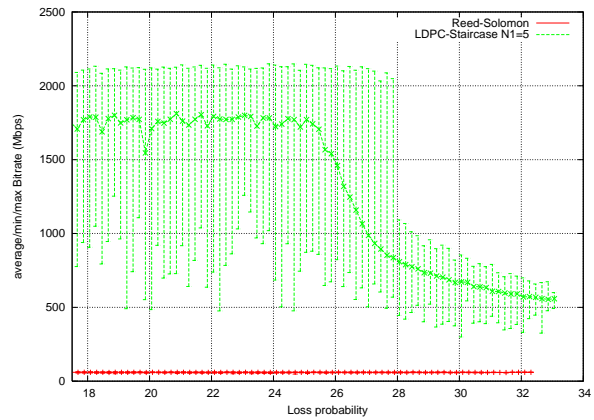
(b) object composed of 1,000 symbols (reminder).
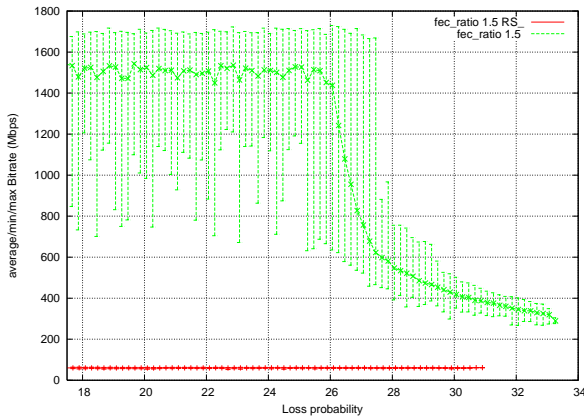
(c) object composed of 2,000 symbols.
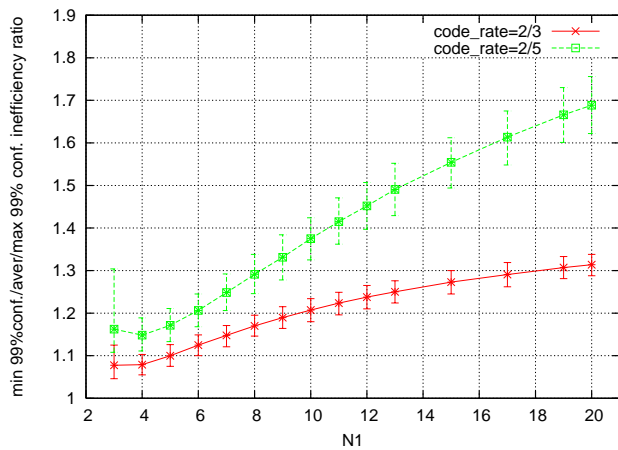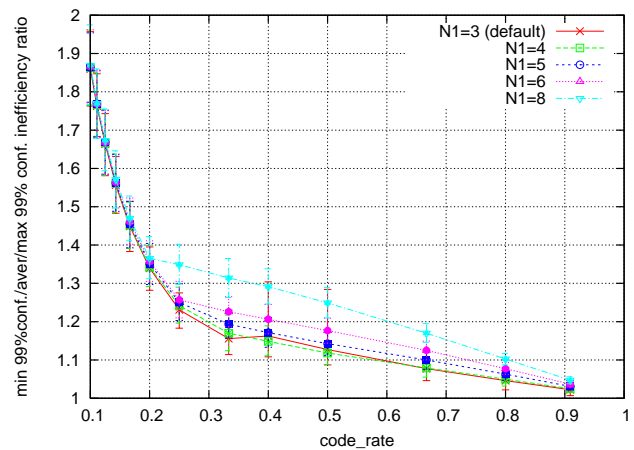

(d) object composed of 4,000 symbols.

Fig. 7.   Bitrate w.r.t. the loss probability for various object sizes (hybrid ID/GE scheme, code rate 2/3, N1=5).

Fig. 8.   ID-only decoder : inefficiency ratio w.r.t. the N1 parameter (object composed of 1,000 symbols).

Fig. 9.   ID-only decoder : inefficiency ratio w.r.t. the code rate (object composed of 1,000 symbols).

the N1 parameter has little impact on the parity check matrix creation algorithm, and therefore on the resulting performances.

In all cases, using a GE (in particular as part of a hybrid scheme) remains efficient even with small objects, composed of a few hundreds of symbols only (which is not true with

ID). Therefore the "number of symbols of an encoding symbol group" parameter (G) specified in [11] (i.e., the number of encoding symbols per packet) should always be set to 1. Said differently, with a GE decoding, there is no benefit in artificially increasing the number of symbols in the block by reducing their size.
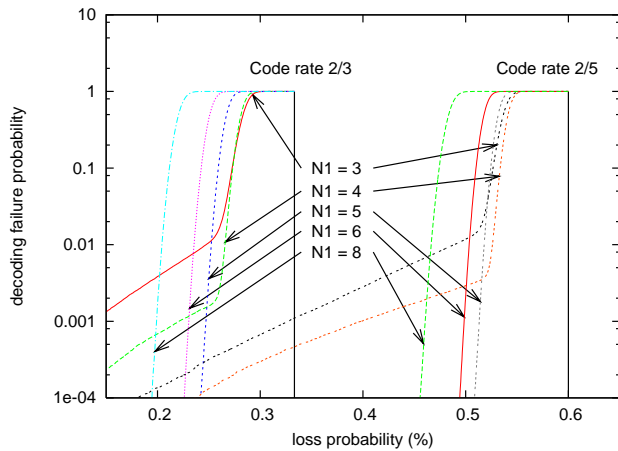
Fig. 10. ID-only decoder : decoding failure probability w.r.t. erasure probability (object composed of 1,000 symbols).

VI. CONCLUSIONS

This work focuses on LDPC-staircase codes that are now an IETF standard (RFC 5170 [11]). We have shown that these codes, when associated to a hybrid ID/GE scheme, approach the performances of ideal codes. It has been made possible by the use of the N1 parameter, that controls the source node degree.

For instance, with a code rate 2/3, an object composed of 1,000 symbols and N1=5, the average decoding overhead amounts to 0.63%, and a decoding failure probability less than 10^{-4} is achieved with an overhead equal to 2.21% (or equivalently loss rate 31.9%).

Even if using a GE scheme has an impact on the decoding complexity, the hybrid decoder remains relatively fast. During our tests (i.e., an object composed of 1,000 symbols, code rate 2/3, N1=5), our LDPC-staircase codec is, on average, between 32.4 times (iterative-only decoding) and 10.2 times (GE-only decoding) faster than the reference Reed-Solomon C codec over $GF(2^8)$.

To conclude, this work shows that the LDPC-staircase codes coupled with a hybrid ID/GE decoder are simple yet efficient codes, that feature at the same time : good erasure recovery capabilities, extremely fast encoding, decoding times that remain at least an order of magnitude faster than that of the reference Reed-Solomon codec and, last but not least, the possibility to trade erasure recovery capabilities for decoding complexity at any time.

VII. ACKNOWLEDGMENT

This work was supported by the French ANR grant No 2006 TCOM 019 (project CAPRI-FEC). The authors also want to thank Valentin Savin and Jérôme Lacan for their comments.

REFERENCES

[1] K. Abdel-Ghaffar and J. Weber. Generalized iterative decoding for linear block codes on the binary erasure channel. *IEEE International Symposium on Information Theory (ISIT'07)*, June 2007.
[2] K. Abdel-Ghaffar and J. Weber. Generalized stopping sets and stopping redundancy. *IEEE Workshop on Information Theory and Applications*, January 2007.
[3] David Burshtein and Gadi Miller. An efficient maximum-likelihood decoding of ldpc codes over the binary erasure channel. *IEEE Transactions on Information Theory*, 50(11) :2837–2844, 2004.
[4] J. Cai, C. Tjhai, M. Tomlinson, M. Ambroze, and M. Ahmed. *A New Non-Iterative Decoding Algorithm for the Erasure Channel : Comparisons with Enhanced Iterative Methods*, March 2005. http ://arxiv.org/abs/cs/0503006.
[5] M. Cunche and V. Roca. Improving the decoding of LDPC codes for the packet erasure channel with a hybrid Zyablov iterative decoding/Gaussian elimination scheme. Research Report 6473, INRIA, March 2008.
[6] D. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, ISBN : 0521642981, 2003.
[7] E. Paolini, G. Liva, B. Matuz, and M. Chiani. Generalized ira erasure correcting codes for hybrid iterative/maximum likelihood decoding. *IEEE Communication Letters*, June 2008.
[8] E. Paolini, M. Varrella, M. Chiani, B. Matuz, and G. Liva. Low complexity ldpc codes with near-optimum performance over the bec. *4th Advanced Satellite Mobile Systems Conference (ASMS'08), http ://aps.arxiv.org/abs/0804.2991*, August 2008.
[9] L. Rizzo. Effective erasure codes for reliable computer communication protocols. *ACM Computer Communication Review*, 27(2), April 1997.
[10] V. Roca and C. Neumann. Design, evaluation and comparison of four large block FEC codecs, LDPC, LDGM, LDGM staircase and LDGM triangle, plus a Reed-Solomon small block FEC codec. Research Report 5225, INRIA, June 2004.
[11] V. Roca, C. Neumann, and D. Furodet. *Low Density Parity Check (LDPC) Staircase and Triangle Forward Error Correction (FEC) Schemes*, June 2008. IETF Request for Comments, RFC 5170.
[12] A. Shokrollahi, S. Lassen, and R. Karp. *Systems and Processes for Decoding Chain Reaction Codes Through Inactivation*, February 2005. U.S. Patent Number 6,856,263.
[13] C. Neumann V. Roca, M. Cunche and J. Labouré. *An Open-Source LDPC Large Block FEC Codec*. URL : http ://planete-bcast.inrialpes.fr/.
[14] V. Zyablov and M. Pinsker. Decoding complexity of low-density codes for transmission in a channel with erasures. *Translated from Problemy Peredachi Informatsii*, 10(1), 1974.