A Simple Byte-Erasure Method for Improved Impulse Immunity in DSL†

Dimitris Toumpakaris*, Wei Yu**, John M. Cioffi*, Daniel Gardan§, and Meryem Ouzzif§
*Department of Electrical Engineering, 350 Serra Mall, Stanford University, Stanford, CA 94305
**Electrical and Computer Engineering Department, 10 King's College Rd, U. of Toronto, Toronto, ON M5S 3G4, Canada
§France Télécom R&D, 2 Av. Pierre Marzin, 22 307 Lannion, France
dtouba@stanford.edu, weiyu@comm.utoronto.ca, cioffi@stanford.edu, {daniel.gardan, meryem.ouzzif}@francetelecom.com

***Abstract*— The data that is transmitted in DSL systems is subject to corruption by Impulse Noise, *i.e.*, noise bursts of high energy that interfere with the transmitted symbols. As DSL data rates increase and crosstalk mitigation techniques become more sophisticated, Impulse Noise limits service in terms of rate or delay. Because of the highly non-stationary nature of Impulse Noise, a combination of interleaving and Reed-Solomon coding is currently used to shield systems from noise bursts. This paper presents a modified Impulse Noise protection algorithm that takes advantage of the improved performance of Reed-Solomon codes when the location of the impaired bytes is known. Without changing the structure of the encoder or the interleaver, it is shown that the delay, or equivalently the overhead due to forward error correction coding, can be reduced without compromising the immunity of the system to impulses. A DMT-VDSL system is used as a particular example of the improvement achieved using byte-erasure.**

I. Introduction

Various sources of interference, such as home appliances, electrical switches or motors can result in noise bursts that appear on the copper lines used by DSL systems to transmit data. Apart from being highly non-stationary, those bursts tend to have energies that are much higher than the static background noise or the crosstalk that is induced by neighboring transmission lines. Therefore, unless some kind of error correction is used, Impulse Noise causes irreversible damage to some of the transmitted symbols, thus leading to significant errors in the received data stream. Obviously, the impact of Impulse Noise depends on the energy, the duration, and the spectral characteristics of the bursts.

In the early stages of DSL deployment, conservative designs with larger-than-needed margins were used, which resulted in significantly lower rates than the ones that can be achieved using more sophisticated techniques. As the DSL market reaches maturity, the demand for higher rates increases and operators realize the added value they can get by efficiently using their local loops, better designs with subsequently lower margins are expected. Although the drive to lower margins has been greatly facilitated by crosstalk mitigation techniques that have appeared recently [1], [2], [3], [4], Impulse Noise still remains a challenge, partly because obtaining an accurate statistical model is not easy. Many researchers have worked on the modeling issue [5], [6], [7], [8], [9]. However, to this day, a consensus on a good Impulse Noise model has not been reached.

Due to the lack of a widely used model, the performance of the proposed algorithm is evaluated using actual impulses measured by France Télécom. Some of the measured impulses are more severe than what is currently specified in the standards [10], [11]. Previous studies of DMT-ADSL and DMT-VDSL systems have shown that some of the impulses that have been measured can be so severe that if the system delay is to be kept within the limits specified by the recommendations, the resulting interleaving depth and Reed-Solomon (RS) code parity are not sufficient to protect those systems [12]. However, the technique used in current systems, which is described in Section II, does not take full advantage of the error-correcting capabilities of RS codes, since the received symbols are decoded without using any knowledge of the location of the corrupted bytes. If these bytes are correctly located they can be "erased" before being passed to the RS decoder. It is well known that in that case the correcting capability of the RS codes can be doubled, without the need for any change in the parity overhead [13]. However, what is really critical is that the corrupted bytes be marked correctly.

This property of RS codes has been used in [14] to develop an erasure method based on unused tones in a DMT-based system. In [15], a method that uses the information provided by the decoder of an inner code to locate potentially corrupted bytes is described. In the present study, it is assumed that the system does not use any inner code, and therefore no a priori information that can be directly passed to the outer FEC decoder is available. It is shown that under certain conditions, erasure of impulse-corrupted bytes can be accurately indicated to the FEC decoder that causes the protection to be nearly doubled, leaving full protection within the delay guidelines of the corresponding standards. This improvement is attained without any change to the existing standards, and solely through improvement of the receiver processing so that potentially corrupted bytes are indicated accurately.

Section II describes the generic block diagram of the Impulse-Noise-protection technique. Section III focuses on interleaving in order to facilitate the understanding of the proposed algorithm that is discussed in Section IV. In Section V the improvement in error-correction performance is demonstrated by simulating the system, and Section VI concludes the paper and presents some topics of possible future research.

II. System Model

The block diagram of the Impulse-Noise-protection system is shown in Fig. 1. In current DSL systems, after having been grouped in blocks of K bytes, the user data are passed to the input of an RS encoder that adds P parity bytes, thus increasing the block size to N. RS Codes are block codes with alphabets in $GF(2^m)$. Current DSL systems use byte-oriented RS codes that operate on $GF(2^8)$, restricting the maximum block size to $N = 255$ bytes. At the receiver, the decoder is able to correct up to $\left\lfloor \frac{P}{2} \right\rfloor$ bytes in error if the location of the errors is unknown, and up to P bytes if it knows the exact location of the corrupted bytes, which are "erased" before being passed to the decoder.

†This work was supported in part by France Télécom R&D.

0-7803-7802-4/03/$17.00 © 2003 IEEE

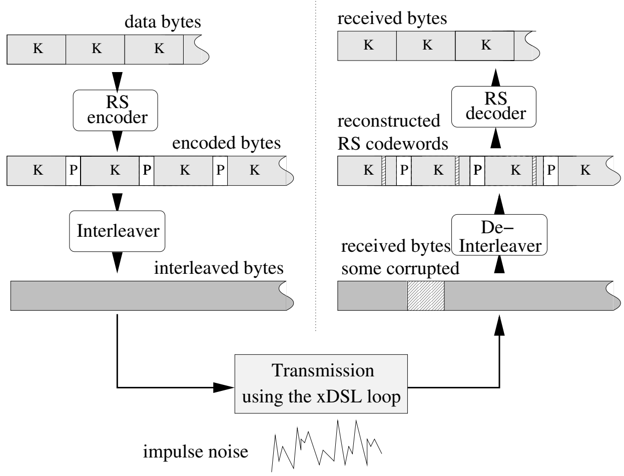

Fig. 1. The system model

More generally, if the number of parity bytes is equal to P, the decoder can correct the bytes in error if $e + 2f \leq P$, where f is the number of errors whose location is unknown, and e is the number of erasures. Assuming correct erasures, if f is such that $2f + e > P$, the decoder will almost certainly declare a failure, in which case the received bytes are left intact. For more details on how RS codes work, the reader is referred to Error-Correction Coding texts, such as [13].

The encoded bytes are then interleaved in order to randomize the error bursts caused by Impulse Noise. Section III examines interleaving more closely, in order to facilitate the understanding of the algorithm proposed in this paper. The interleaved stream is then sent to the DSL system for transmission. The details on how the stream will be transmitted depend on the specific standard that is used. Note that the Impulse Noise protection in currently deployed DSL systems is completely decoupled from transmission and the system treats the interleaver output as user data.

On the receiver side, the data are demodulated and sent to a deinterleaver that has exactly the same parameters as the interleaver at the transmitter. The bytes at the output of the deinterleaver form the original RS blocks, with some bytes possibly corrupted. The RS decoder attempts to reconstruct the original bytes and succeeds provided that there is no block with more than $\lfloor \frac{P}{2} \rfloor$ corrupted bytes, since, as mentioned in the introduction, no error-localization algorithm – and hence no erasure decoding – is used in current implementations.

III. Interleaving

Fig. 1 shows that errors caused by Impulse Noise occur in bursts, irrespective of the particular modulation scheme that is used. RS parity is not by itself sufficient to provide error-correction, since when a noise burst occurs, the number of corrupted bytes in a block can easily exceed the number of bytes that can be corrected by the code. The combination of interleaving and deinterleaving overcomes this problem by spreading the error bursts. After deinterleaving, the errors due to an impulse will be distributed on a small number of bytes of several RS blocks instead of many consecutive bytes of a small number of blocks. The improvement in error correction due to interleaving is frequently called interleaving gain.

However, the interleaving gain is achieved at the expense of increased end-to-end delay. Since bytes of a given block are

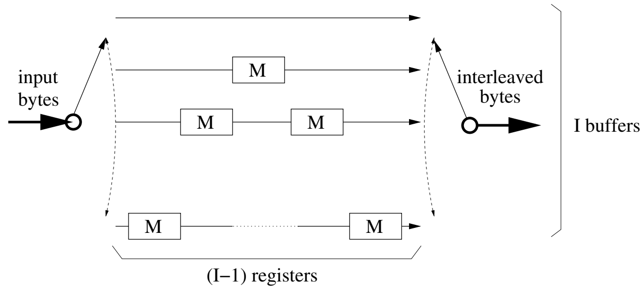

Fig. 2. Triangular Interleaver

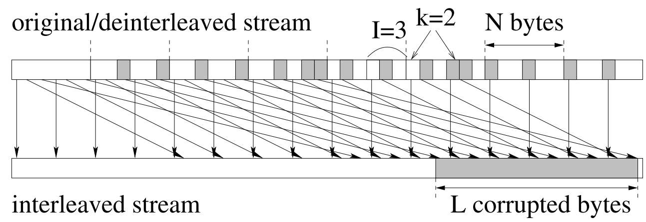

Fig. 3. Inverse Mapping of corrupted bytes to user stream when using a generalized triangular interleaver

dispersed, in order to pass that block to the RS decoder the deinterleaver will have to wait for all the bytes of the block to arrive. Therefore, the delay experienced by the last byte of the block will determine the overall end-to-end delay. Obviously, larger interleaver depths that improve the performance of the system against impulses result in longer end-to-end delays. The objective is to minimize the delay without deteriorating the immunity of the system to noise bursts.

A. The generalized triangular interleaver

The interleaver proposed for VDSL systems [16] is the so-called generalized triangular convolutional interleaver. Its block diagram is shown in Fig. 2. Before interleaving, data are grouped in blocks of size I bytes, where $N = kI$, k an integer, and $N = K + P$ is the size of the RS codewords. M is a parameter that determines the interleaver depth. Each byte of each block of I bytes is sent to a buffer with different delay. The first byte does not experience any delay. The second byte is shifted within the register of length M each time a new byte enters its buffer, *i.e*, each time the second byte of a block is sent to the interleaver. Therefore, the total time that it remains in the interleaver is equal to IM. Proceeding in the same way, it can be seen that the i-th byte of each block is delayed by $(i-1)IM$. Hence, the end-to-end delay when using the generalized triangular interleaver is equal to $(I-1)IM$.

Typical values that are used are $k = 8$ for $(N, K) = (240, 224)$, which results in blocks of size $I = 30$, and $k = 4$ for $(N, K) = (144, 128)$, which yields blocks of size $I = 36$. In this case, since the first byte of each block of size I is not delayed, k bytes of each RS codeword are not delayed. Another k bytes are delayed by IM, and so on, the largest delay of $(I-1)IM$ being experienced by the k I-th bytes of each sub-block of a RS codeword. An example of the mapping when using the generalized triangular interleaver is given in Fig. 3, where $N = 6$, $k = 2$ and $I = 3$ to simplify the diagram. Note that the value of M that has to be used for a given level of protection depends on I.

B. Calculation of the required interleaver depth

Suppose that $\lfloor \frac{P}{2} \rfloor \geq k$, and that the i-th byte of an RS block gets mapped to a location in the interleaved stream that is corrupted by Impulse Noise. The delay of this byte is $((i-1) \; mod \; I)IM$. In the worst case, all the other $k-1$ bytes that experience the same delay will fall on the corrupted area of the interleaved stream. Suppose now that the next group of k bytes in the same RS block whose delay is $(i \; mod \; I)IM$ also falls in the corrupted area of the interleaved stream. Then, up to $2k$ bytes of the original block may be in error. By induction, if the error-correction capability of the RS code is equal to t bytes, then the system can tolerate up to $\lfloor \frac{t}{k} \rfloor$ such "groups" of bytes to fall on the corrupted bytes. Hence, the difference in the delay of bytes that are more than $\lfloor \frac{t}{k} \rfloor$ groups apart should be larger than the length of the corrupted area. Therefore, if the corrupted area has a length of L bytes, we must have

$$\left\lfloor \frac{t}{k} \right\rfloor IM > L \Rightarrow M > \frac{L}{\lfloor \frac{t}{k} \rfloor I} \Rightarrow M_{min} = \left\lceil \frac{L}{\lfloor \frac{t}{k} \rfloor I} \right\rceil .$$

Note that the value of M depends on the block size I. When the location of the corrupted bytes is unknown, $t = \lfloor \frac{P}{2} \rfloor$, whereas if there is a way to provide erasures to the RS decoder, $t = P$, and therefore the delay can be reduced by a factor up to 2.

Consider the $(m+1)$-th RS codeword whose two last bytes – with associated delays $(I-2)IM$ and $(I-1)IM$ bytes – are mapped to the corrupted area, as shown in Fig. 4. Suppose that no bytes from the previous (m-th) RS codeword are mapped into the noise area. Then,

$$\begin{aligned} mN + (I-1)IM < (m+1)N - 1 + (I-2)IM \Rightarrow \\ N - 1 - IM > 0 \Rightarrow kI - 1 - IM > 0 \Rightarrow \\ (k-M)I - 1 > 0. \end{aligned}$$

Note that if instead of the penultimate byte of the $(m+1)$-th block we had considered any other byte experiencing delay of $(I-2)IM$ bytes, the required value for k with respect to M would be even larger. For VDSL, the values of M are larger than k. Consider for example a VDSL system with transmission rate equal to 20 Mbps. Suppose that a noise burst corrupts data corresponding to 500 μs. Then, in the worst case, $L = 5 \times 10^{-4} \times 20 \times 10^{6} = 10000$ bits $= 1250$ bytes will be corrupted. When a $(240, 224)$ RS code is used with $I = 30$ and $k = 8$, $M = 42 > k = 8$. Similarly, it can be shown that $M > k$ when a $(144, 128)$ code is used in a typical VDSL system. This means that for a system designed to deal with error bursts of length equal to 500 μs, the number of corrupted "groups" of k bytes cannot increase by more than 1 between consecutive RS codewords. Therefore, RS codewords with all $\lfloor \frac{t}{k} \rfloor$ "groups" of k bytes potentially corrupted will be preceded by RS codewords with a maximum of $1, 2, \ldots, \lfloor \frac{t}{k} \rfloor - 1$ "vulnerable" groups. This observation, *i.e.*, that the number of affected groups increases progressively in increments of 1 before we reach the maximum of $\lfloor \frac{t}{k} \rfloor$ will be a key part of the algorithm of Section IV.

The above is proven rigorously in [17]. Moreover, when using a generic convolutional interleaver, an equivalent condition can be derived: $\frac{L}{N} > \frac{P}{\lfloor \frac{P}{2} \rfloor}$, which also holds for VDSL systems.

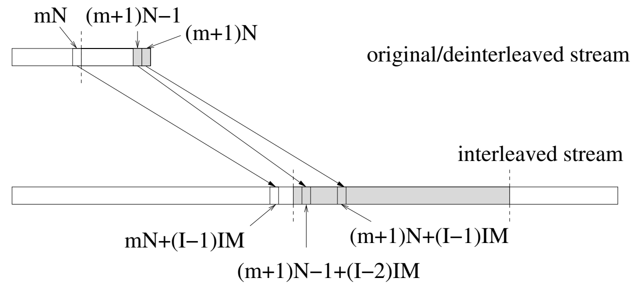

Fig. 4. Derivation of the minimum required value for M

C. Interleaver Parameters for the DMT-VDSL case

The remainder of this paper considers the case of DMT-VDSL, although the observations and the results can be applied to other DSL systems as long as the condition $M > k$ is satisfied. In DMT-VDSL, the duration of each symbol is equal to 250 μs. A noise burst of 500 μs, which is the maximum duration specified in the standards [10], [11], will affect up to 3 DMT symbols depending on the alignment of the FFT and the temporal occurrence of the Impulse Noise burst. In general, the receiver's FFT spreads the impulse so that some tones of the DMT symbol will not be affected. However, some of the impulses measured in actual local loops have very high energies and can corrupt most of the bytes of each symbol. Moreover, it has been reported that in some cases the duration of these impulses can exceed 500 μs. Therefore, the simulations of Section V focus on the design of a system that can correct up to 4 DMT symbols in the interleaved buffer.

Suppose that each DMT symbol carries B bytes. Then, in the worst case, up to $4B$ consecutive bytes (belonging to 4 consecutive DMT symbols) will be corrupted. As shown in Section III, M has to satisfy

$$M > \frac{L}{\lfloor \frac{t}{k} \rfloor I} = \frac{4B}{\lfloor \frac{t}{k} \rfloor I} \Rightarrow M_{min} = \left\lceil \frac{4B}{\lfloor \frac{t}{k} \rfloor I} \right\rceil .$$

Then, the end-to-end delay is $d = MI(I-1) > \frac{4B(I-1)}{\lfloor \frac{t}{k} \rfloor}$ bytes. To convert the delay to seconds, recall that it takes 250 μs to transmit B bytes. Therefore,

$$d_{min} = \left\lceil \left\lceil \frac{4B}{\lfloor \frac{t}{k} \rfloor I} \right\rceil \frac{I(I-1)}{B} \right\rceil 250 \; \mu s > \frac{(I-1)}{\lfloor \frac{t}{k} \rfloor} \text{ ms.}$$

Table I enumerates the lower bound of the minimum delay which is required in order to protect VDSL systems that use $(240, 224)$ and $(144, 128)$ RS codes from the worst-case impulses. It is clear that unless a way is found to provide erasures to the RS decoder, use of $(240, 224)$ codes is not an option in systems that are subject to severe impulses, if full impulse protection is required. Even when using the larger-overhead $(144, 128)$ codes, the delay is near the limit of 20 ms that is set in the standards.

IV. The Algorithm

The modified algorithm that improves Impulse Noise protection is first presented, followed by a discussion of the different steps.

TABLE I
MINIMUM DELAY FOR IMPULSE NOISE PROTECTION IN DMT-VDSL

code (N,K)	I	k		delay
$(240,224)$	30	8	no erasures	29 ms
			erasures	14.5 ms
$(144,128)$	36	4	no erasures	17.5 ms
			erasures	8.75 ms

Assumptions

- The exact mapping between the DMT symbols of the interleaved buffer and the RS codewords of the deinterleaved stream is known.
- A noise burst affects up to n DMT symbols
- $M > k$, $M > \frac{nB}{\lfloor \frac{P}{k} \rfloor I}$, and $\lfloor \frac{P}{2} \rfloor \geq k$.
- The interarrival times between impulse bursts are larger than the end-to-end delay. Hence, errors in an RS block are due to one impulse event only. This is an assumption that is also made in the standards.

The algorithm

for *each RS codeword*
  decode the codeword without using erasures
  if *decoding is successful*
    find if there were any corrupted bytes
      by comparing the input to the output
      or using the codeword syndrome
    if *errors were found*
      mark the corresponding DMT symbols as corrupted,
        as long as there are no more than n consecutive
        marked DMT symbols
    end
  else *in the case of decoding failure*
    if *the number of marked DMT symbols r is less than n*
      and *there are bytes that are mapped to an unmarked DMT symbol that is less than n DMT symbols ahead of the first marked DMT symbol of the sequence*
      mark this DMT symbol as well
    end
    erase the bytes that correspond to the marked
      DMT symbols
    decode using the erased bytes
  end
end

To improve Impulse Noise protection the algorithm takes advantage of the fact that before encountering RS blocks with the maximum number of bytes corrupted, there will be indication of the occurrence of an impulse because of errors in previous RS codewords as it was shown in Section III-B. If the number of errors is less than or equal to $\lfloor \frac{P}{2} \rfloor$, those errors can be corrected and therefore the locations of the bytes in error can be determined. From these erroneous bytes, one can trace the DMT symbols of the interleaved stream that those bytes belong to assuming synchronization between the DSL system and the error-correcting system. Then, these DMT symbols can be marked as corrupted.

As long as the RS decoder can correct the bytes in the blocks without using knowledge of their location, the standard method is used. However, when it fails to decode a block, the information on the corrupted DMT symbols is used and all the bytes that come from the corrupted DMT symbols will be erased. Obviously, the algorithm works well provided that M is large enough so that erasures can work, and that the maximum number of corrupted DMT symbols is correctly estimated a priori.

A tricky situation arises when some bytes of a codeword that cannot be decoded unless erasures are used, belong to DMT symbols that have not been marked yet. Under the assumption of consecutive corrupted DMT symbols, those bytes can be erased and correction can be made as long as the total number of marked consecutive symbols is less than or equal to n, so that M be sufficient for erasure decoding to work.

The algorithm is robust to random errors due to background noise and crosstalk, since such events will only corrupt one byte and they will not divert the algorithm into the erasure mode. A problem can only occur when a random error falls on an RS codeword that is also corrupted by Impulse Noise. In this case, it is possible for the algorithm to enter the erasure mode and fail to decode correctly a certain number of RS blocks. However, the probability that a random error coincides with a severe noise burst is very low for typical target bit error rates (which are usually of the order of 10^{-7}). Another common practice is to erase fewer than P bytes in order to be able to locate random errors with the remaining parity bytes. For example, one could design a system with a slightly larger end-to-end delay where the maximum number of erasures in each RS block is equal to $P - 2$, allocating 2 parity bytes to the correction of one corrupted byte at an unknown location. This would lead to a slightly increased end-to-end delay, but in the same time it would significantly improve the robustness of the system.

Finally, although in some cases an RS block has to be decoded twice, the modified algorithm does not have a major impact on the overall delay of the system since the time required for the decoding of RS codes is small compared to the delay due to interleaving. Even if the decoding delay is an issue, a pipelining structure could be used that passes the codeword through two consecutive decoders, from which the second one is only used when the algorithm gets into the decoding failure mode, else it simply forwards the decoded block of the first decoder to the output.

V. SIMULATION RESULTS

Simulated transmission on a DMT-VDSL system with different Impulse Noise bursts is used to verify the improvement in error correction when using the modified algorithm. A data buffer is constructed and then encoded using $(N,K) = (240,224)$ or $(144,128)$. Then the data are transmitted to the downstream direction using a DMT-VDSL system with parameters shown in Table II. The transmission is impaired by a deterministic impulse burst that is superimposed on the received modulated data. The received data are then demodulated and decoded using the standard mechanism, as well as the modified algorithm of Section IV. Three representative impulses provided by France Télécom are used. The first two represent the worst-case scenario for which the system is designed. They are of long duration and high energy and corrupt a large number of bytes of 4 consecutive DMT symbols. The third impulse is shorter (115 μs) and for simulation purposes is superimposed on two DMT symbols.

Table III compares the end-to-end delay that is required in order to protect the system from each of the three impulses. It also gives the corresponding value of M for the target rate of Table II. Note some deviations from the minimum delay

TABLE II

DMT-VDSL PARAMETERS USED FOR THE SIMULATIONS

parameter	value
transmission	Downstream FTTEx
rate (to upper layer)	23.168 Mbps
tones + cyclic prefix	4096 + 320 (complex)
frequency spacing	4.3125 kHz
symbol error probability	10^{-7}
band plan	998
PSD mask	Pex.P2.LT.M1 of [11]
maximum power	11.5 dB
margin	6 dB
coding gain	3.86 dB for $(240, 224)$ code, 4.22 dB for $(144, 128)$ code
crosstalk noise	Model E of ETSI

TABLE III

MINIMUM DELAY FOR IMPULSE NOISE PROTECTION

noise number	code (N, K)	delay (no erasures)	M (no erasures)	delay (erasures)	M (erasures)
1	$(240, 224)$	27.75 ms	106	14.75 ms	56
	$(144, 128)$	16.75 ms	44	9.25 ms	24
2	$(240, 224)$	29.25 ms	112	14.75 ms	56
	$(144, 128)$	17.75 ms	47	9.25 ms	24
3	$(240, 224)$	12 ms	46	7.5 ms	28
	$(144, 128)$	6.25 ms	16	4.75 ms	12

values of Table I. The lower values occur when not all bytes of the affected DMT symbols are corrupted. The larger values are due to the fact that the delays of Table I are on the lower bound for worst-case noise.

As expected, the end-to-end delay is significantly reduced without the need to compromise the level of Impulse Noise protection or reduce the length of the DSL loop. The reduction is larger (approaching a factor of 2) for the worst-case noises, and smaller for milder impulses or impulses of shorter duration, since the algorithm marks a whole DMT symbol when it detects that some of its bytes are corrupted. However, usually the system design targets worst-case impulses since the standards require total Impulse Noise immunity at the interleaved channel, so the resulting end-to-end delay can be reduced by a factor of approximately 2. If it is tolerated that some impulses corrupt the transmitted data, methods that refine the localization of the erroneous bytes need to be employed. Section VI contains a brief discussion on such methods that are currently being investigated.

VI. CONCLUSION

By taking advantage of the nature of the mapping of the interleaved buffer to the original RS codewords, a modified algorithm was developed that improves the protection against Impulse Noise. The assumptions made are realistic and are also based on actual measurements by operators. The proposed algorithm does not require any change of the encoder and interleaver structure and can substitute the current decoders whenever more robust Impulse Noise protection is required. Moreover, it works even when no inner code is used by the DSL system. The expected performance of the algorithm was derived by considering the effect of the combination of interleaving and deinterleaving and the theoretical predictions were verified by simulating Impulse Noise events that impair a DMT-VDSL system.

In systems employing an inner code (such as Wei's 4-dimensional Trellis code used in DMT systems) the double pass required by the algorithm of section IV can be reduced to a single erasure correction decoding stage, by using the information available at the decoder of the inner code. This is the subject of [15]. Regardless of whether an inner code is used or not, the localization of the corrupted area can also be achieved by monitoring the mean-square distance between the allowed and the received constellation points, as described in [17]. This last method that further improves localization could potentially enable better cancellation of RF interference due to HAM operators that appears outside the standardized amateur bands due to circuit and channel nonlinearities. This is a topic of current research.

REFERENCES

[1] G. Ginis and J.M. Cioffi, "Vectored Transmission for Digital Subscriber Line Systems," *IEEE J. on Select. Areas Commun. special issue on twisted-pair transmission*, vol. 20, no. 5, pp. 1085-1104, June 2002.

[2] W. Yu, G. Ginis and J.M. Cioffi, "An Adaptive Multiuser Power Control Algorithm for VDSL," *IEEE J. on Select. Areas Commun. special issue on twisted-pair transmission*, vol. 20, no. 5, pp. 1105-1115, June 2002.

[3] M. L. Honig, K. Steiglitz and B. Gopinath, "Multichannel signal processing for data communications in the presence of crosstalk," *IEEE Trans. Commun.*, vol. 38, no. 4, pp. 551–558, Apr. 1990.

[4] M. L. Honig, P. Crespo and K. Steiglitz, "Suppression of near- and far-end crosstalk by linear pre- and post-Filtering," *IEEE J. on Select. Areas Commun.*, vol 10, no. 3, pp. 614–629, Apr 1992.

[5] I. Mann, S. McLaughlin, and W. Henkel, "Impulse Generation with appropriate amplitude, length, inter-arrival, and spectral characteristics," *IEEE J. on Select. Areas Commun.*, vol. 20, no. 5, pp. 1–8, May 2002.

[6] R. Kirkby, D. B. Levey and S. McLaughlin, "Statistics of Impulse Noise," ETSI UK TM6, TD18, TD19, TD20, TD21. Edinburgh, UK, 1999.

[7] I. Mann, S. McLaughlin and D. B. Levey, "A new statistics for Impulse Noise measurement," ETSI UK TM6, TD 55. Amsterdam, The Netherlands, 1999.

[8] B. Rolland, D. Bardouil, F. Clérot, D. Collobert, D. Gardan, J. Le Roux, "Classification du bruit impulsif par méthode d'apprentissage non supervisé," Internal report RP/FTR&D/7133, France Télécom, December 2000.

[9] D. Collobert *et al.*, "Proposal for Impulse Noise Classification," ETSI TM6, TD12. Sophia Antipolis, France, April 2002.

[10] T1E1.4/2000-009R3. VDSL Technical Specification Part 1. Functional Requirements and Common Specification.

[11] ETSI TS 101 270–1 v1.2.1 (1999–10). Transmission and Multiplexing (TM); Access transmission systems on metallic access cables; Very high speed Digital Subscriber Lines (VDSL); Part 1: Functional Requirements.

[12] W. Yu, D. Toumpakaris and J. M. Cioffi, "The Performance of Asymmetric Digital Subscriber Lines in an Impulse Noise Environment ," *submitted to the IEEE Transactions on Communications*.

[13] S. Wicker, *Error Control Systems for Digital Communication and Storage*, Prentice Hall, 1995.

[14] F. Sjöberg, "The Zipper Duplex Method in Very High-Speed Digital Subscriber Lines," Ph.D thesis, Luleå University of Technology, April 2000.

[15] D. Toumpakaris, W. Yu, J. M. Cioffi, D. Gardan, and M. Ouzzif, "A Byte-Erasure Method for Improved Impulse Immunity in DSL Systems using Soft Information from an Inner Code" *accepted for presentation in ICC 2003*, to appear.

[16] T1E1.4/2000-013R4. Very-high bit-rate Digital Subscriber Lines (VDSL) Metallic Interface, Part 3: Technical Specification of a Multi-Carrier Modulation Transceiver.

[17] D. Toumpakaris, J. M. Cioffi, and D. Gardan, "Reduced Delay Protection of DSL Systems against Non-Static Disturbances", *Submitted to IEEE Trans. Commun.*, unpublished.