TSG-RAN Working Group 1 meeting #2 **TSGR1#2(99) 069**
Yokohama 22-25, February 1999

Source : **CANON CRF**[1]

Title : **1 dimensional algebraic interleavers for turbo codes (AL-C): description, complexity and summary of performances**

1 Description of algebraic interleavers

1.1 General description

For a complete description, see [1], [2] and mainly [3].
We consider Parallel Concatenated Convolutional Code (PCCC) with 8 state constituent encoders as described below.
The transfer function of the first constituent encoder is:

$$G_1(D) = [1, n_1(D)/d(D)].$$

And the transfer function of the second constituent encoder is:

$$G_2(D) = [n_2(D)/d(D)]$$

Where

$$d(D) = 1+D^2+D^3$$
$$n_1(D) = 1+D+D^3$$
$$n_2(D) = 1+D+D^2+D^3$$

(Note: It is not mandatory that both encoders are different. Nevertheless, it has been observed that algebraic interleavers give better performances when the constituent encoders are different.)

The conventional method of trellis termination is used in which the tail bits are taken from the shift register feedback of the first constituent encoder after all information bits are encoded. Tail bits are added after the encoding of information bits.

Then, the complete sequence including information bits **and tail bits** is interleaved with an algebraic interleaver.
Let $a(x)=\Sigma_{i=0 \text{ to } n-1}\, a_i x^i$ represent a sequence of n binary digits a_i.

The algebraic interleaver is defined so that the permuted sequence is equal to:

$$a^*(x)=\Sigma_{i=0 \text{ to } n-1}\, a_i x^{e.i} \textit{ modulo } (x^n-1)$$

where e is a constant depending on n.

To determine e, we take all the power of 2 modulo n and the appropriate value is selected by evaluation based on code distance and simulation.
(E.g. for n = 329, possible values of e are: 2 4 8 16 32 64 128 256 183 37 74 148 296 263 197 65 130 260 191 53 106 212 95 190 51 102 204 79 158 316 303 277 225 121 242 155 310 291 253 177 25 50 100 200 71 142 284 239 149 298 267 205 81 162 324 319 309 289 249 169 9 18 36 72 144 288 247 165. For these possible values of e, simulations provided e = 32 as best candidate.)

[1] Contact Persons: Claude Le Dantec, Philippe Le Bars, Philippe Piret, CANON CRF
Rue de la Touche Lambert, 35517 Cesson Sévigné Cedex, France
Tel : +33 (0) 2 99 87 68 77 Fax: +33 (0) 2 99 84 11 30 email:ledantec@crf.canon.fr , lebars@crf.canon.fr, piret@crf.canon.fr

Algebraic interleavers are defined for n which must be an odd multiple of 7.

Within conditions of simulations defined by the Turbo codes ad hoc group for next meeting:

- $n = 329$ (i.e. 326 information bits plus 3 tail bits), $e = 32$
- $n = 637$ (i.e. 634 information bits plus 3 tail bits), $e = 246$
- $n = 5117$ (i.e. 5114 information bits plus 3 tail bits), $e = 3957$.

Finally, **the interleaved sequence is encoded with the second constituent encoder without any additional bits**. In these conditions, the second encoder memories will also return to all zero state due to some properties of algebraic interleavers.

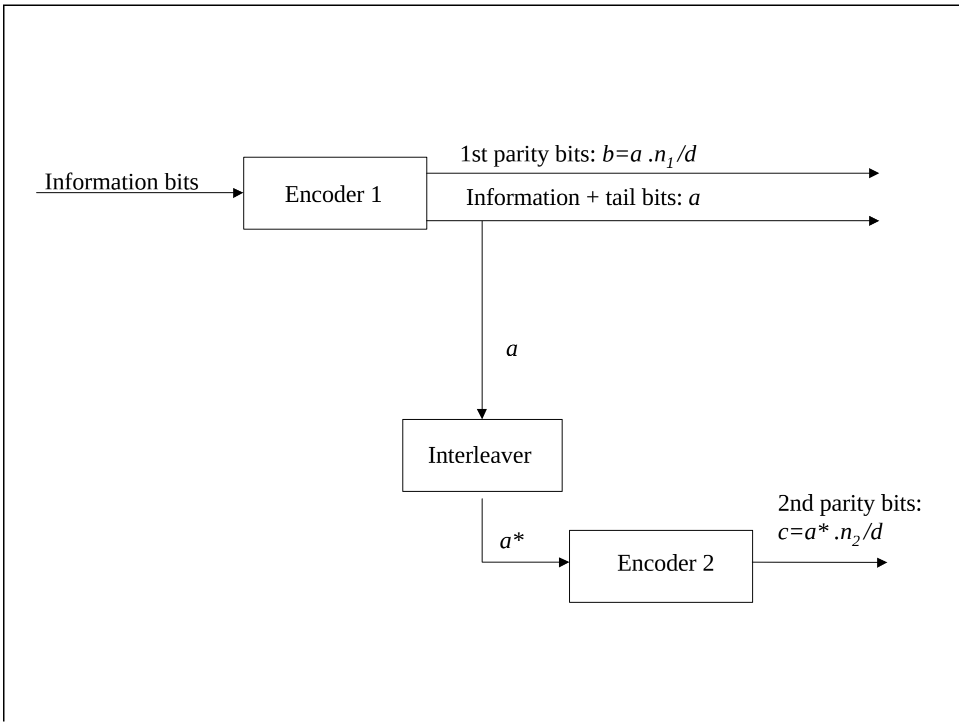

Fig. 1: Turbo encoder scheme with algebraic encoders

1.2 Implementation of algebraic interleavers

We emphasise that one advantage of algebraic interleaver is simplicity, not only for interleaver but also for de-interleaver.
Let i represent the position of any bit of a sequence at the interleaver input.
After interleaving, this bit will be listed in a position defined by the function *interleave*(i) defined with *interleave*(i) = $i.e$ modulo n.

This can also be iteratively obtained in a very simple way with the following algorithm:

```
interleave(0) = 0
For i = 1 to n-1
        interleave(i) = interleave(i-1) + e
        if interleaver(i) ≥ n
                then interleaver(i) = interleaver(i) – n
        endif
End For
```

The desinterleaver can be implemented in the same way as the interleaver. For a given interleaver of size n defined with a parameter e as described above, the corresponding desinterleaver is defined with a parameter e^{-1} so that $e.e^{-1} = 1$ modulo n. It is only necessary to store one value (e) per interleaver length.

As example,

- for n = 329, e = 32 and e^{-1} = 72
- for n = 637, e = 246 and e^{-1} = 246
- for n = 5117, e = 3957 and e^{-1} = 4645.

1.3 Adaptation to any length of algebraic interleavers.

In the above, Algebraic Interleavers have been described with a length which is an odd multiple of 7. Nevertheless, it is possible to adapt algebraic interleavers so that a turbo coder using these interleavers can match any size of information block.

For this, the first encoder is kept as described above: this encoder provides a sequence *a* corresponding to the input bits plus tail bits that terminates the trellis.
Let *k* be the length of this sequence.
Before interleaving, this sequence is completed with (*n*-*k*) null bits to build a sequence *b* of size *n* which is an odd multiple of 7. Then, the complete sequence *b* is interleaved with an algebraic interleaver and encoded with the second constituent encoder. In these conditions, the second encoder memories will return to all zero state.
The decoding as usual is done using the consideration that trellis is terminated and that the null bits padded to the sequence *a* are perfectly known.

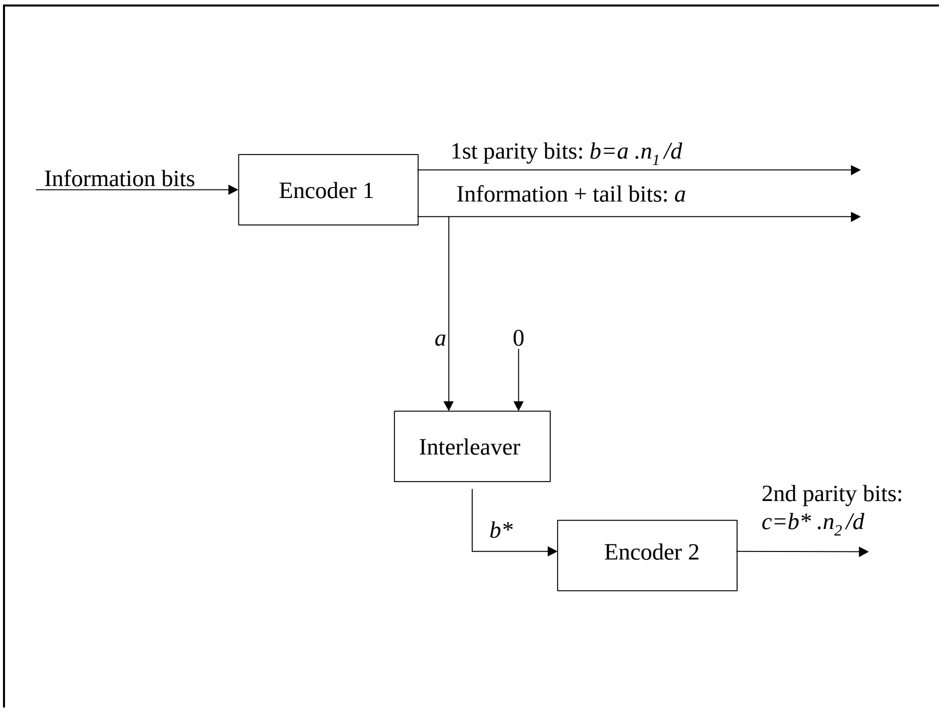

Fig. 2: Turbo encoder scheme with algebraic encoders for any size of input sequence

For example, with an input sequence of size 640 bits, 3 tail bits are added by the first encoder to ensure the trellis termination. Thus, k = 643, and eight additional null bits are added before interleaving. Then, the interleaver size is here equal to 651 bits. Thus, 1929 bits (information + tail bits: 643 bits; 1st parity: 643 bits; 2nd parity: 651 bits) are transmitted.

2 Complexity figures

2.1 ASIC implementation complexity

The algebraic interleavers from Canon (AL-C) are fully defined with the parameter *e* depending on *n*. Hence, the required look up table is very small: 9 bits if $n < 512$, 10 bits if $n < 1024$ and 13 bits if $4096 \leq n < 8192$.
(I.e. a total of 22 bits for n # 320, 640 and 5120).
Furthermore, algebraic interleavers from Canon and Nortel have similar gate count and power dissipation. (See [5] for details).

Using figures and assumptions given in [5], we can summary the ASIC implementation complexity for interleaver size close to 320, 640 and 5120 bits as following. (Complexity factor regarding AL-C is put into brackets).

Interleaver	**Gate Count**	**Power dissipation**	**Look Up Table**
AL-C	220	220	22
AL-N	220 (x1)	220 (x1)	120 (x5)
GF	5240 (x23)	500 (x2)	2064 (x93)
MIL	#5900 (x27)	#2000 (x9)	3904 (x177)

2.2 DSP implementation complexity

The following C function can be implemented to build an algebraic interleaver of size 637. Its rough translation to DSP instruction is listed on the right side.

```
{   e = 246                                         1 mov
    interleave(0) = 0;                              1 mov
    for (i = 1; i < n; i++)                         1 dec and test (x n-2)
    {   interleave(i) = interleave(i-1) + e;        1 add (x n-2)
        if (interleaver(i) ≥ n)                     1 conditional equate (x n-2)
        {   interleaver(i) = interleaver(i) – n;    1 add (x n-2)
        }
    }
}
```

Total: 2+(n-2)x4= 2+(635x4) =2540 operations.

Using figures and assumptions given in [5], we can summary the DSP implementation complexity for interleaver size close to 640 bits as following:

Interleaver	**AL-C**	**AL-N**	**MIL**	**GF**
Total weights	2540	4576 (x1.8)	14919 (x5.8)	#36000 (x14)

3 Performances

We can sum up the simulation results for a decoding process with 4 iterations:
UMTS SMG2 L1 channel (software provided by Nokia) and the channel estimator provided by HNS (the best one as defined in UMTS SMG2 L1 meeting #9) (See [4])

Service	Speed	E_b/N_o (given BER)			
		AL-C	**MIL**	**GF**	**AL-N**
d=10ms 32kbps	3kmph	5.2 dB (10^{-5}) 5.45 dB (10^{-6})	5.3 dB (10^{-5}) 5.55 dB (10^{-6})	5.3 dB (10^{-5})	5.3 dB (10^{-5})
	30kmph	5.4 dB (2.10^{-6})	5.6 dB (2.10^{-6})	5.6 dB (2.10^{-6})	5.7 dB (2.10^{-6})
d=10ms 64kbps	3kmph	4.2 dB (10^{-5})	4.15 dB (10^{-5})	4.15 dB (10^{-5})	4.15 dB (10^{-5})
	30kmph	4.3 dB (10^{-5})	4.2 dB (10^{-5})	4.2 dB (10^{-5})	4.2 dB (10^{-5})
d=80ms 64kbps	3kmph	3.95 dB (4.10^{-5})	3.7 dB (4.10^{-5})	3.7 dB (4.10^{-5})	3.8 dB (4.10^{-5})
	30kmph	3.9 dB (2.10^{-5})	3.6 dB (2.10^{-5})	3.6 dB (2.10^{-5})	3.7 dB (2.10^{-5})

Service	Speed	E_b/N_o (given FER)			
		AL-C	**MIL**	**GF**	**AL-N**
d=10ms 32kbps	3kmph	5.15 dB (2.10^{-3})	5.3 dB (2.10^{-3})	5.3 dB (2.10^{-3})	5.3 dB (2.10^{-3})
	30kmph	5.4 dB (5.10^{-4})	5.6 dB (5.10^{-4})	5.6 dB (5.10^{-4})	5.6 dB (5.10^{-4})
d=10ms 64kbps	3kmph	4.15 dB (7.10^{-3})	4.15 dB (7.10^{-3})	4.15 dB (7.10^{-3})	4.15 dB (7.10^{-3})
	30kmph	4.4 dB (2.10^{-3})	4.4 dB (2.10^{-3})	4.4 dB (2.10^{-3})	4.4 dB (2.10^{-3})
d=80ms 64kbps	3kmph	3.9 dB (10^{-1})	3.75 dB (10^{-1})	3.7 dB (10^{-1})	3.8 dB (10^{-1})
	30kmph	3.75 dB (10^{-1})	3.65 dB (10^{-1})	3.55 dB (10^{-1})	3.65 dB (10^{-1})

4 Conclusion

Thus, we can conclude that all selected interleavers give similar results with a 4 iteration decoding process. Regarding complexity whether DSP or ASIC, AL-C have simpler implementation than other interleavers.

5 References

[1] SMG2 UMTS L1 Tdoc 571/98, “Algebraic interleavers for turbo codes”, CANON CRF
[2] SMG2 UMTS L1 Tdoc 674/98, “Algebraic interleavers for turbo codes: Simulation results”, CANON CRF

[3] SMG2 UMTS L1 Tdoc 721/98, “Algebraic interleavers for turbo codes: Precisions on complexity and interleaver generation”, CANON CRF
[4] SMG2 UMTS L1 Tdoc 038/99, “Algebraic interleavers for turbo codes: Further simulation results”, CANON CRF
[5] SMG2 UMTS L1 Tdoc 051/99, “Low complexity algebraic interleaver for UTRA Turbo Codes”, Nortel Networks