Results and discussion: In Fig. 4 optical signals in channels 0, 4 and 10, detected at the corresponding photodiodes of 0, 4 and 10, respectively, are shown against time as an example. The data were treated by the software, LabVIEW (National Instruments Co. Ltd.). We can see from Fig. 2*b* and Fig. 4 that the centre-diode, 0, observes just 14 optical peaky signals corresponding to the 14 slit array, while 4 and 10 detect the similar two main peaks at the same time, which correspond to the 4th and 11th slits having the same slit angle in the slit array, as counted from the slit at the far left. We observe some undesirable optical signals originating from the other diffracted-light patterns passing photodiodes 4 and 10 with time as shown in Fig. 4. When the laser beam spot becomes large enough to irradiate adjacent slits at the same time, these undesirable optical signals are observed. They can, however, be distinguished from the aimed diffracted-light pattern, since the aimed optical signals should be detected simultaneously at 0, and at symmetrically positioned photodiodes, such as 4 and 10; furthermore its intensity should be stronger than that of undesirable optical signals. Each optical peaky signal has a different width, as shown in the channel 0 optical signals against time in Fig. 4, which is caused by the difference of passing time through the centre-diode, 0, due to the difference in the rotational angle of the slit.

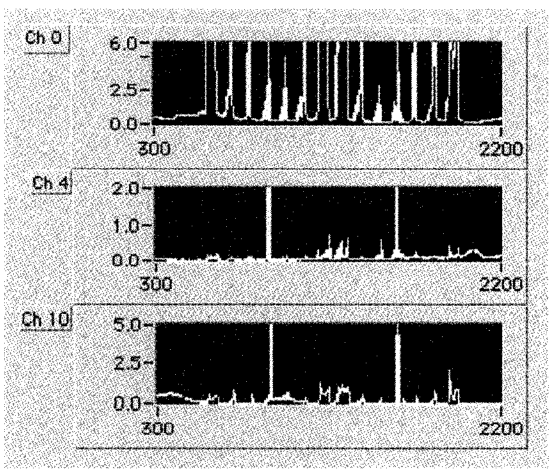

Fig. 4 *Optical signals against time in channels 0, 4 and 10, detected at the corresponding photodiodes in the photodiode array of 0, 4 and 10, respectively, due to the diffracted-light patterns at the 14 slit array shown in Fig. 2b*

In this first experiment using the prototype memory system, the rotational slit pitch-angle of 30° is adopted and the transmitted diffraction-light-pattern from the slit is used instead of the reflected pattern, to enable a better understanding and to demonstrate this new concept of optical memory. The rotational pitch-angle of 30° corresponds to a 6 valued recording. The optical memory of a 36 valued recording will easily be realised by adopting the rotational pitch-angle of 5° between slits and by increasing the number of photodiodes in the array; this will lead to more than 5 times the usual binary recording density of optical memory.

This multiple-valued recording of a transmitted diffracted-light pattern from the slit can be applied to the reflected pattern at the slender pit as for that in CD, because the transmitted and the reflected diffraction are theoretically the same. If we use this memory as a CD-ROM, a very small pit with the prescribed angle with the recording track can be formed by the use of electron beam lithography; this promises a much higher memory density.

Conclusion: A new concept in optical memory is proposed and demonstrated, based on diffracted-light pattern recognition, giving multiple-valued information at a recording spot-area of an optical disk instead of the usual binary optical memory, promising higher density optical memory. It uses an He-He laser as the light source, a slit array with various angles with the recording track for generation of the transmitted diffracted-light pattern at the slit and a photodiode array for pattern recognition, especially the angle of the pattern. The recorded rotational slit-angle with the recording track gives multiple-valued information in this new system and the rotational slit-angle is determined by the photodiode array. By shortening the laser wavelength and adopting a high NA optical lens for reading and writing, a higher density optical memory will be realised.

Acknowledgment: We are grateful to Y. Takahashi for his help with experiments.

24 March 1997
Electronics Letters Online No: 19970608

M. Kimura and T. Takahira (*Electrical Engineering, Faculty of Engineering, Tohoku-Gakuin University, 1-13, Chuo-1, Tagajo, Miyagi, 985, Japan*)

References

1 PASCHOTTA, R., FIELDER, K., KURZ, P., HENKING, R., SCHILLER, S., and MLYNEK, J.: '82% efficient continuous-wave frequency doubling of 1.06μm with a monolithic MgO:LiNbO$_3$ resonator', *Opt. Lett.*, 1994, **19**, pp. 1325–1327

2 MIYAGAWA, N., GOTOH, Y., OHNO, E., NISHIUCHI, K., and AKAHIRA, N.: 'Land and groove recording for high track density on phase-change optical disks', *Jpn. J. Appl. Phys.*, 1993, **32**, pp. 5324–5328

Non-iterative optimum super-trellis decoding of turbo codes

M. Breiling and L. Hanzo

Indexing terms: Turbo codes, Decoding

An optimum non-iterative turbo decoder is proposed that is highly parallelisable and out-performs the maximum a-posteriori decoder.

Introduction: Turbo codes discovered by Berrou *et al.* [1] exhibit near-Shannonian BER against channel SNR performance at the cost of increased complexity due to iterative decoding. Both the power and the complexity of turbo codes accrues from the scrambling performed by the interleaver at the input of its second encoder.

Algorithm: Consider the encoder structure depicted in Fig. 1, where the interleaved information sequence $\mathbf{u}^{(2)}$ is a permutation of the original sequence $\mathbf{u}$. For the sake of illustration, we assume a two-column block-interleaver, and hence

$$\mathbf{u}^{(2)} = (u_1; u_3; u_5; u_7; ...; u_2; u_4; u_6; ...) \qquad (1)$$

The encoding of the original sequence can be modelled by a path through a trellis, where each branch corresponds to a transition associated with a bit of the information sequence $\mathbf{u}$ being a binary 0 or 1. The encoding of the interleaved sequence $\mathbf{u}^{(2)}$ can be modelled in the same way, the only difference being the order of the information sequence bits. When decoding, the maximum likelihood path through the trellis has to be found. Viterbi proposed a dynamic programming algorithm in [2], that keeps the complexity relatively low without making any concessions to the optimality in the sense of finding the maximum likelihood path.

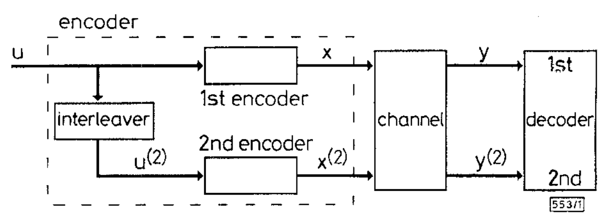

Fig. 1 *Simplified turbo codec*

The problem associated with decoding turbo codes is that their structure is described by two trellises, one for $\mathbf{u}$ and one for $\mathbf{u}^{(2)}$. When we want to decode the bits contained in $\mathbf{u}$ serially, we end up with a trellis for $\mathbf{u}^{(2)}$ which is split into several disjoint sections. Fig. 2 shows the trellis segments of the two trellises which are associated with the first seven data bits in $\mathbf{u}$, using the interleaver of eqn. 1.

Conventional turbo decoding techniques therefore decode the two trellises for **u** and $\mathbf{u}^{(2)}$ separately. They pass soft-information about the estimated $\hat{\mathbf{u}}$ from the first decoder to the second one, which then uses this additional information for decoding its trellis and in turn, it generates another sequence of soft-information about $\hat{\mathbf{u}}$, and passes this back to the first encoder. This iteration can be executed several times, yielding ever better results for a higher number of repetitions. Although the maximum likely information sequence $\hat{\mathbf{u}}_{opt}$ can be approached by this method, it is rarely exactly identified.

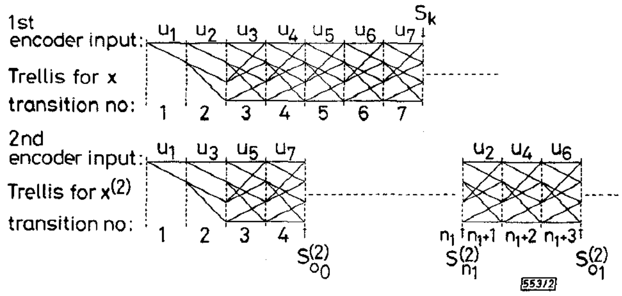

Fig. 2 *Trellis transitions associated with bits u_1...u_7*

To answer some questions concerning optimality and convergence speed of the above mentioned iterative algorithm, in [3] Benedetto and Montorsi posed, among others, the following questions:

(i) What is the performance of the maximum likelihood decoder?

(ii) How close are the known iterative algorithms to it?

(iii) How important are recursive convolutional component codes for turbo codes?

These three questions are to be answered in this Letter.

In contrast to conventional iterative decoders, our proposed method aims at finding the maximum likelihood information sequence without iterations. To make this possible, a way has to be found to transform the two split trellises into a kind of super-trellis which a 'Viterbi-like' algorithm could be used for. This super-trellis is found, when we establish superstates S_k^* by amalgamating the states in all the open trellis ends of Fig. 2, yielding in our example:

$$S_k^* = \left(S_k; S^{(2)}_{o_0}; S^{(2)}_{n_1}; S^{(2)}_{o_1}\right) \qquad (2)$$

where S_k denotes the state at the end of the left-hand section in the upper trellis, and $S^{(2)}_{o_0}$, $S^{(2)}_{n_1}$ and $S^{(2)}_{o_1}$ denote the state at the end of the left-hand section, as well as at the start and the end of the right-hand section of the lower trellis, respectively, all after decoding stage k. As S_k^* takes on all legitimate values s^*, it represents the possible states of a super-trellis, which can then be decoded by a conventional Viterbi decoding approach. We have proven that this method always identifies the optimum codeword.

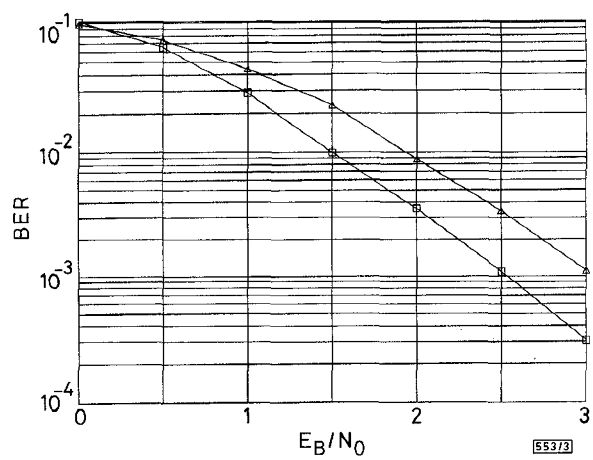

Fig. 3 *Performance of proposed non-iterative decoder in contrast to MAP decoder (1000 error events pr. point)*

□ flat decoder
△ turbo (MAP) decoder

Results: The super-trellis based Viterbi algorithm was implemented, and as Fig. 3 shows, its performance is ~0.5dB better than that of a maximum a-posteriory (MAP) decoder [4] with 16 iterations for a 3 × 333 bits block-interleaver (RSC-component code with rate $R = 1/2$, memory $m = 2$ and the generator polynomials $g_0 = 7$ and $g_1 = 5$). We have hence answered questions (i) and (ii): The iterative algorithm is very close to the optimum, but it does not converge. A simulation with a five column interleaver (which increases the decoder's complexity by a factor of 256) supports this assumption. A simulation executed with a non-recursive code (systematic codes are not necessary for the presented ML decoder) seems furthermore to back up the conclusion in [3] that non-recursive component codes are less performing than recursive codes. A turbo code with the same specification as in Fig. 3 but using a non-recursive component code achieved a BER of 2.26 × 10^{-2} at an SNR of 1.5dB.

The number of valid superstates increases exponentially with the number of open trellis ends preventing us from using our proposed technique for more complex interleavers. Future work must therefore concentrate on exploiting the section-wise structure of the super-trellis and find less complex, but sub-optimal non-iterative approaches.

Furthermore, the notion of the underlying super-trellis structure of turbo codes is doubtlessly also very interesting from the information theoretical point of view, possibly allowing us to answer the remaining questions in [3] and other yet unsolved problems concerning turbo codes.

10 March 1997
Electronics Letters Online No: 19970596

M. Breiling and L. Hanzo (*Department of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, United Kingdom*)

E-mail: lh@ecs.soton.ac.uk

References

1 BERROU, C., GLAVIEUX, A., and THITIMAJSHIMA, P.: 'Near Shannon limit error-correcting coding and decoding: Turbo-codes'. Proc. ICC, 1993, pp. 1064–1070

2 VITERBI, A.J.: 'Error bounds for convolutional codes and an asymptotically optimum decoding algorithm', *IEEE Trans. Info. Theory*, 1967, **13**, pp. 260–269

3 BENEDETTO, S., and MONTORSI, G.: 'Unveiling turbo codes: Some results on parallel concatenated coding schemes', *IEEE Trans. Info. Theory*, 1996, **42**, pp. 409–428

4 ROBERTSON, P., VILLEBRUN, E., and HOEHER, P.: 'A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain'. IEEE Int. Conf. Commun., 1995, pp. 1009–1013

Scalable 2D arrays of noise sources for stochastic retina

A. Dupret, B. Granado and P. Garda

Indexing terms: Gaussian noise, Cellular automata

The design of scalable 2D arrays of Gaussian white noise sources is addressed. Two original circuits based on cellular automata are described, and the choice of the cellular automaton transition function is discussed.

Introduction: Electronic retina performing stochastic image processing at video rate require a 2D array of white noise sources [2]. Suitable random generators were demonstrated [4], but they require a sophisticated optical setup. In this Letter, two new methods based on homogeneous boolean cellular automata are introduced for the design of fully electronic 2D random generators.

A homogeneous boolean cellular automaton comprises a set of identical cells. Each cell has a Boolean state computed as a result of a transition function whose inputs are the previous states of the neighbouring cells. With some transition functions, chaotic dynamics of the cellular automaton are observed [5].