Factorizable modulo *M* parallel architecture for DVB-S2 LDPC decoding

Marco Gomes, Gabriel Falcão, Vitor Silva, Vitor Ferreira, Alexandre Sengo and Miguel Falcão*
Instituto de Telecomunicações, Pólo II da Universidade de Coimbra, 3030-290 Coimbra, Portugal
*Chipidea Microelectrónica S.A., Rua Frederico Ulrich, n. 2650, 4470-605 Moreira da Maia, Portugal
e-mail: marco@co.it.pt, gff@co.it.pt, vitor@co.it.pt, vitorhugo@co.it.pt, sengo@co.it.pt, mfalcao@chipidea.com

Abstract — State-of-the-art decoders for DVB-S2 low-density parity-check (LDPC) codes explore semi-parallel architectures based on the periodicity $M = 360$ factor of the special type of LDPC-IRA codes adopted. This paper addresses the generalization of a well known hardware *M*-kernel parallel structure and proposes an efficient partitioning by any factor of *M*, without addressing overhead and keeping unchanged the efficient message memory mapping scheme. Our method provides a simple and efficient way to reduce the decoder complexity. Synthesizing the decoder for an FPGA from Xilinx shows a minimum throughput above the minimal 90Mbps.

I. INTRODUCTION

The recent Digital Video Satellite Broadcast Standard (DVB-S2) [1] [2] has adopted a powerful FEC scheme based on the serial concatenation of BCH and Low Density Parity Check (LDPC) codes. This new FEC structure, combined with the adoption of high order modulations (QPSK, 8PSK, 16APSK and 32APSK), is able to provide capacity gains of about 30% over the previous DVB-S standard [2], with the LDPC codes playing a fundamental role in this raise of performance.

LDPC codes are linear block codes defined by sparse parity-check matrices [3] [4] [5], **H** and, usually, represented by Tanner graphs [6]. A Tanner graph is a bi-partite graph formed by two types of nodes. Check nodes (v^{C}), one per each code constraint, and bit nodes one per each codeword bit (information and parity, respectively, v^{I} and v^{P}), with the connection edges between them being given by **H**.

They are decoded using low complexity iterative belief propagation algorithms operating over the Tanner graph description [7]. However, a major drawback is their high encoding complexity caused by the fact that the generator matrix, **G**, is, in general, not sparse. In order to overcome this problem, DVB-S2 standard has adopted a special class of LDPC codes, with linear encoding complexity, known by Irregular Repeat-Accumulate (IRA) [8] [9].

An important issue in the design of LDPC encoder and decoder architectures for DVB-S2 is the fact that the standard supports two different frame lengths (16200 bits for low delay applications and 64800 bits otherwise) and a set of different code rates ($1/4$, $1/3$, $2/5$, $1/2$, $3/5$, $2/3$, $3/4$, $4/5$, $5/6$, $8/9$ and $9/10$) for both frame lengths and different modulation schemes [1] [9]. For each mode of operation is defined a different LDPC code and, although they share a similar structure and properties, this still poses an enormous challenge on the development of an encoder and a decoder fully compliant with all operating modes.

The decoder state-of-the-art is based on a flexible partial parallel architecture that explores the $M = 360$ periodicity nature of DVB-S2 LDPC codes [10]. Although capable of providing a throughput far above from the minimum mandatory rate of $90\,Mbps$, this architecture requires a huge ASIC area of $22.74\,mm^2$ on a ST Microelectronics $0.13\,\mu m$ technology, mainly due to the high number (360) of computation kernels or functional units (FU) and the wide length of the barrel shifter. In order to decrease the number of computation kernels to only 45 FU's and to reduce the length of the barrel shifter, an alternative solution was proposed [11] which uses a re-structured version of **H**. As a consequence, this approach increases the complexity of the DVB-S2 de-interleaver and doubles (almost) the input memory in terms of [10].

In this paper we generalize the architecture [10] and surpass its disadvantages. We will show that it is possible to reduce the number of computation kernels to any integer factor of $M = 360$, without addressing overhead and keep unchanged the efficient message memory mapping scheme [10]. Our strategy also reduces the length of the barrel shifter by the same factor and considerably simplifies the routing problem. The throughput is reduced by the same factor but this does not represent a real problem since the architecture [10] is able to provide a throughput far above from the mandatory minimum rate. Thus, we provide a simple and efficient method to reduce the decoder complexity without loosing the throughput goals.

The next section briefly describes DVB-S2 LDPC-IRA codes. Section III addresses the LDPC decoding for DVB-S2 using a partial parallel architecture and its generalization by sub-sampling it by a factor of *M*. Synthesis results are presented in section IV and final conclusions are pointed out in section V.

II. DVB-S2 LDPC-IRA CODES

The new DVB-S2 [1] [9] standard adopted a special class of LDPC codes known by IRA codes [8] as the main solution for the FEC system. An IRA code is characterized by a parity check matrix, **H**, of the form,

CALTECH - EXHIBIT 2007

$$\mathbf{H}_{(n-k)\times n} = \left[\mathbf{A}_{(n-k)\times k} \;\middle|\; \mathbf{B}_{(n-k)\times(n-k)} \right]$$
$$= \left[\begin{array}{cccc|cccccc} a_{00} & a_{01} & \cdots & a_{0,k-1} & 1 & 0 & \cdots & \cdots & \cdots & 0 \\ a_{10} & a_{11} & \cdots & a_{1,k-1} & 1 & 1 & 0 & & & \vdots \\ \vdots & & & \vdots & 0 & 1 & 1 & \ddots & & \vdots \\ \vdots & & & \vdots & \vdots & \ddots & \ddots & \ddots & 0 & \vdots \\ a_{n-k-2,0} & a_{n-k-2,1} & \cdots & a_{n-k-2,k-1} & \vdots & & \ddots & 1 & 1 & 0 \\ a_{n-k-1,0} & a_{n-k-1,1} & \cdots & a_{n-k-1,k-1} & 0 & \dots & \dots & 0 & 1 & 1 \end{array}\right], \qquad (1)$$

where $\mathbf{B}$ is a staircase lower triangular matrix. By restricting $\mathbf{A}$ to be sparse, it is obtained an LDPC-IRA code [9].

The $\mathbf{H}$ matrices of the DVB-S2 LDPC codes have other properties beyond being of IRA type. Some periodicity constraints were put on the pseudo-random design of the $\mathbf{A}$ matrices, which allows a significant reduction on the storage requirement without code performance loss.

The matrix $\mathbf{A}$ construction technique is based on dividing the v^{I} nodes in disjoint groups of M consecutive ones. All the v^{I} nodes of a group l should have the same weight, w_l, and it is only necessary to choose the v^{C} nodes that connect to the first v^{I} of the group in order to specify the v^{C} nodes that connect to each one of the remaining $M-1$ nodes. The connection choice for the first element of group l is pseudo-random with the restrictions that the resulting LDPC code is cycle-4 free, the number of length 6 cycles is the shortest possible and all the v^{C} nodes must connect to the same number of v^{I} nodes.

Denoting by, $r_1, r_2, \dots, r_{w_l}$, the indices of the v^{C} nodes that connect to the first v^{I} of group l, the indices of the v^{C} nodes that connect to v_i^{I}, with $0 \le i \le M-1$, of group l can be obtained by,

$$\begin{array}{l} \{(r_1 + i\times q)\,\mathrm{mod}(n-k),\, (r_2 + i\times q)\,\mathrm{mod}(n-k), \dots, \\ (r_{w_l} + i\times q)\,\mathrm{mod}(n-k)\}, \end{array} \qquad (2)$$

with $q = (n-k)/M$ and $M = 360$ (a common factor for all DVB-S2 supported codes).

Another property of matrix $\mathbf{A}$ is that for each supported code, there are a set of groups of v^{I} nodes of constant weight $w > 3$ (w is code dependent) and the remaining have all weight 3.

III. DVB-S2 LDPC DECODING

The huge dimensions of the LDPC-IRA codes adopted by the DVB-S2 standard, turns impractical the adoption of a fully parallel architecture that maps the Tanner graph structure [12]. Besides that, such solution is code dependent, which means that is required a different full parallel decoder for each code defined in the standard.

Best known solutions are based on highly vectorized partial parallel architectures [10] [11], that explore the particular characteristics of the DVB-S2 LDPC-IRA codes, namely, the periodic nature ($M = 360$) shared by all the codes. One solution was proposed in [10], whose architecture uses M functional units working in parallel. In this paper we will show that it is possible to reduce the number of functional units by any integer factor of M, without addressing overhead, keeping unchanged its efficient memory mapping scheme. Our approach does not only surpass the architecture [10] disadvantages, but also makes the architecture flexible and easy reconfigurable according with the decoder constraints.

A. Modulo M parallel architecture

As previously described, DVB-S2 adopted a special class of structured LDPC-IRA codes with the properties stated in (2). This turns possible the simultaneous processing of v^{I} and v^{C} node sets, whose indices are given by,

$$\begin{array}{ll} \mathrm{C}^{(c)} = \{c, c+1, \cdots, c+M-1\}, \text{ with, } c \bmod M = 0, \text{ and} & \\ \mathrm{R}^{(r)} = \{r, r+q, r+2q, \cdots, r+(M-1)q\}, \text{ with, } 0 \le r \le q-1, & (3) \end{array}$$

respectively, (the superscript is the index of the first element of the set and, ‘r’ and ‘c’ mean row and column of **H**), which significantly simplifies the decoder control. In fact, according to (2), if $v_{\tilde{c}}^{\mathrm{I}}$ is connected to v_r^{C}, then, $v_{r+i\times q}^{\mathrm{C}}$, with $0 \le i \le M-1$, will be connected to, $v_{c+(\tilde{c}-c+i)\bmod M}^{\mathrm{I}}$, where, $c = M \times (\tilde{c} \operatorname{div} M)$ is the index of the first v^{I} of the group $\mathrm{C}^{(c)}$ to which $v_{\tilde{c}}^{\mathrm{I}}$ belongs.

The architecture shown in Fig. 1 is based on M functional units (FU) working in parallel with shared control signals [12], that process both v^{C} (in check mode) and v^{I} nodes (in bit mode) in a flooding schedule manner [13] [14]. Attending to the zigzag connectivity between v^{P} and v^{C} nodes, they are updated jointly in check mode following a horizontal schedule approach [15]. A detailed description of the FU operation can be found in [12].

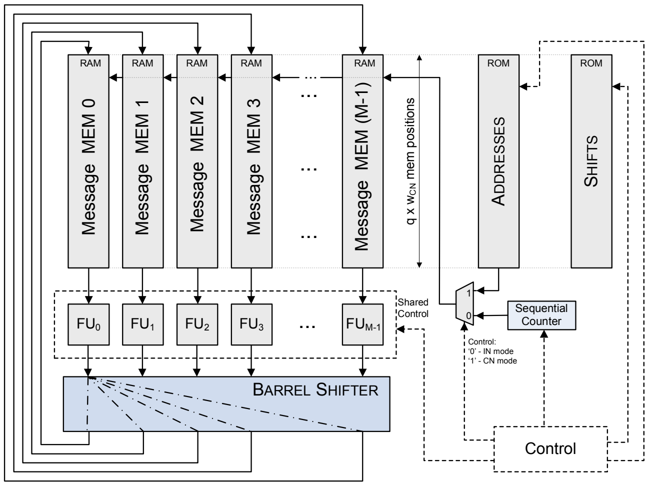

Figure 1. Modulo M parallel architecture for DVB-S2 LDPC decoding.

Memory mapping and shuffling mechanism

As mentioned before, a single FU unit is shared by a constant number of v^{I}, v^{C} and v^{P} nodes (the last two are processed jointly), depending on the code length and rate. More precisely, for a (n,k) DVB-S2 LDPC-IRA code, the FU$_i$, with $0 \le i \le M-1$, updates sequentially in bit mode the $\nu^{\mathrm{I}}_{\{i, i+M, i+2\times M, \cdots, i+(\alpha-1)\times M\}}$ nodes, with $\alpha = k/M$. In check mode, the same FU updates the $\nu^{\mathrm{C}}_{\{j, j+1, \cdots, j+q-1\}}$ and $\nu^{\mathrm{P}}_{\{j, j+1, \cdots, j+q-1\}}$ nodes, with $j = i \times q$. This guarantees that when processing simultaneously the group $\mathrm{C}^{(c)}$, the computed messages have as destination a set $\mathrm{R}^{(r)}$, where each one of them will be processed by a different FU. Considering (2), the new computed messages only need to be right rotated to be handled by the correct ν^{C} nodes. The same happens when processing each $\mathrm{R}^{(r)}$ set, where according to (2), the right rotation must be reversed in order to the new computed messages have as destination the exact ν^{I} nodes. The shuffling network (barrel shifter) is responsible for the correct message exchange between ν^{C} and ν^{I} nodes, emulating the code Tanner graph. The shift values stored in ROM (Fig. 1) can be easily obtained from the annexes B and C of DVB-S2 standard tables [1].

The messages sent along the Tanner graph edges are stored in RAM (see Fig. 1). If we adopt a sequential RAM access in bit mode, then, the access in check mode must be indexed or vice-versa. Both options are valid, so, without loss of generalization, we assume sequential access in bit mode. Denoting by, $\mathbf{r}_i = [r_{i1}\ r_{i2} \cdots r_{iw_i}]^T$, the vector of ν^{C} node indices connected to the ν^{I}_i node of weight, w_i, then, the message memory mapping can be obtained using the following matrix,

$$\mathbf{R} = \begin{bmatrix} \mathbf{r}_0 & \mathbf{r}_1 & \cdots & \mathbf{r}_{M-1} \\ \mathbf{r}_M & \mathbf{r}_{M+1} & \cdots & \mathbf{r}_{2M-1} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{r}_{(\alpha-1)\times M} & \mathbf{r}_{(\alpha-1)\times M+1} & \cdots & \mathbf{r}_{\alpha\times M-1} \end{bmatrix}_{(q\times w_C)\times M} , \qquad (4)$$

where, w_C, is a code constant (ν^{C} weight is $w_C + 2$, except for the first one (1)).

In order to process each $\mathrm{R}^{(r)}$ set in check mode, the required memory addresses can be obtained by finding the matrix **R** rows where the index r appears.

B. Sub-sampling by a factor of M

The simplicity of the shuffling mechanism and the efficient memory mapping scheme, constitute the major strengths of the architecture just described [10]. However, the high number of FU's and the long width of the barrel shifter require a huge silicon area. Since this architecture is able to provide a throughput far above from the minimum mandatory rate of $90\,Mbps$, we may reduce the number of FU's. In fact, we will show that this can be done by any factor of M .

Let be $L, N \in \mathbb{N}$ factors of M , with, $M = L \times N$, and consider a $\mathrm{C}^{(c)}$ set (3). This group can be decomposed by down-sampling in L subgroups as:

$$\begin{aligned} \mathrm{C}^{(c)}_0 &= \{c, c+L, c+2L, \cdots, c+(N-1)\times L\} \\ \mathrm{C}^{(c)}_1 &= \{c+1, c+1+L, c+1+2L, \cdots, c+1+(N-1)\times L\} \\ &\vdots \\ \mathrm{C}^{(c)}_{L-1} &= \{c+L-1, c+2L-1, c+3L-1, \cdots, c+N\times L-1\} \end{aligned} . \qquad (5)$$

Each sub-group, $\mathrm{C}^{(c)}_\gamma$, with $0 \le \gamma \le L-1$, can be described in terms of the first node of the subgroup (2), $\nu^{\mathrm{I}}_{c+\gamma}$. If ν^{C}_r is connected to the first information node of the subgroup, $C^{(c)}_\gamma$, then, $\nu^{\mathrm{C}}_{(r+i\times L\times q)\,\mathrm{mod}(n-k)}$ is connect to the i-th ν^{I} node, with $0 \le i \le N-1$, of the referred subgroup.

Equally, the same down-sampling process by L can be done on each $\mathrm{R}^{(r)}$ group as:

$$\begin{aligned} \mathrm{R}^{(r)}_0 &= \{r, r+L\times q, r+2L\times q, \cdots, r+(N-1)\times L\times q\} \\ \mathrm{R}^{(r)}_1 &= \{r+q, r+(L+1)\times q, r+(2L+1)\times q, \cdots, r+((N-1)\times L+1)\times q\} \\ &\vdots \\ \mathrm{R}^{(r)}_{L-1} &= \{r+(L-1)\times q, r+(2L-1)\times q, r+(3L-1)\times q, \cdots, r+(N\times L-1)\times q\} \end{aligned} \qquad (6)$$

and, in a similar way, each subgroup, $\mathrm{R}^{(r)}_\beta$, with $0 \le \beta \le L-1$, can be described just in terms of the first element, $\nu^{\mathrm{C}}_{r+\beta\times q}$. If $\nu^{\mathrm{I}}_{\tilde{c}}$ is connected to the first node of sub-set $\mathrm{R}^{(r)}_\beta$, then, $\nu^{\mathrm{I}}_{c+((\tilde{c}-c+i\times L)\,\mathrm{mod}\,M)}$, with $c = M \times (\tilde{c}\ \mathrm{div}\ M)$, is connected to the i-th ν^{C} , with $0 \le i \le N-1$, of the considered subgroup.

From the framework just described in (5) and (6), we conclude that the down-sampling approach preserves the key modulo *M* properties and, thus, we can process individually each $\mathrm{C}^{(c)}_\gamma$ and $\mathrm{R}^{(r)}_\beta$ subgroup and the same architecture [10] can be used with only *N* processing units as shown in Fig. 2. In fact, when processing simultaneously a group $\mathrm{C}^{(c)}_\gamma$, the computed messages have as destination a set $\mathrm{R}^{(\mathrm{r})}_\beta$ and, vice-versa.

Memory mapping and shuffling mechanism

The down-sampling strategy allows a linear reduction (by a factor of *L*) of the hardware resources occupied by the FU's blocks, reduces significantly the complexity of the barrel shifter ($O(N \log_2 N)$) and simplifies the routing problem. Yet, at first glance, it may seem that this strategy implies an increase by L in the size of the system ROM (*Shifts* and *Addresses*). Fortunately, if we know the properties of the subgroups $\mathrm{C}^{(c)}_0$ and $\mathrm{R}^{(r)}_0$, we automatically know the properties of the remaining subgroups, $\mathrm{C}^{(c)}_\gamma$ and $\mathrm{R}^{(r)}_\beta$ respectively, with $0 \le \gamma, \beta \le N-1$. By a proper message memory mapping based on a convenient reshape by *L* of the matrix **R** (4), we can keep unchanged the size of the system ROM and compute on the fly the new shifts and addresses values as functions of the ones stored in the ROM of Fig. 2, i.e., for all $\mathrm{C}^{(c)}_0$ and $\mathrm{R}^{(r)}_0$ groups.

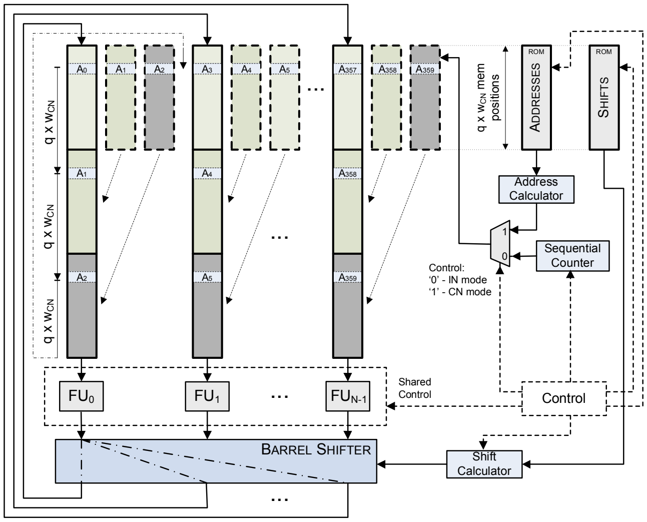

Figure 2. Factorizable modulo *M* parallel architecture for DVB-S2 LDPC decoding.

For the configuration shown in Fig. 2, each FU$_i$, with $0 \leq i \leq N-1$, is now responsible for processing $L \times \alpha$ information nodes in the following order

$$\begin{aligned}&\{i, i+M, i+2M, \cdots, i+(\alpha-1)M\ ;\\ &i+1, i+1+M,\ \cdots, i+1+(\alpha-1)M\ ;\ \ \cdots\ \ ;\ \ \ ,\\ &i+L-1, i+L-1+M,\ \cdots, i+L-1+(\alpha-1)M\,\}\end{aligned} \tag{7}$$

and $L \times q$ check and parity nodes, $\{j, j+1, \cdots, j+L \times q-1\}$, with $j = i \times L \times q$.

IV. SYNTHESIS RESULTS

The architecture of Fig. 2 was synthesized on Virtex-II Pro FPGAs (XC2VP) from Xilinx. For XC2VPxx family it is necessary to use a factor $L=8$ (45 FU's) due to internal memory limitations. In fact, synthesis results show that it is mandatory to use at least the FPGA XC2VP50 in order to guarantee the minimum memory resources required to implement all code rates and lengths. However, this particular choice uses less than 50% of the FPGA available slices. Using external memory, it would be possible to choose the lower cost FPGA XC2VP30.

The XC2VP100 FPGA allows the implementation of the architecture of Fig. 2 with 90 FUs, which doubles the throughput.

V. CONCLUSIONS

This paper addresses the generalization of a state-of-the-art *M*-kernel parallel structure for LDPC-IRA DVB-S2 decoding, for any integer factor of $M = 360$ by mean of sub-sampling, keeping unchanged the efficient message memory mapping structure without addressing overheads. This architecture proves to be flexible and easily reconfigurable according to the decoder constraints and represents a trade off between silicon area and decoder throughput.

Synthesis results show that the implementation of a complete LDPC-IRA DVB-S2 decoder is possible with 45 functional units for Xilinx XC2VP FPGAs family.

REFERENCES

[1] ETSI, *Digital video broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broad-band satellite applications*: EN 302 307 V1. 1.1, 2005.

[2] A. Morello and V. Mignone, "DVB-S2: The second generation standard for satellite broad-band services," *Proceedings of the IEEE*, vol. 94, pp. 210-227, 2006.

[3] R. G. Gallager, "Low-Density Parity-Check Codes," *Ire Transactions on Information Theory*, vol. 8, pp. 21-&, 1962.

[4] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Transactions on Information Theory*, vol. 45, pp. 399-431, 1999.

[5] S. Y. Chung, G. D. Forney, T. J. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Communications Letters*, vol. 5, pp. 58-60, 2001.

[6] R. M. Tanner, "A Recursive Approach to Low Complexity Codes," *IEEE Transactions on Information Theory*, vol. 27, pp. 533-547, 1981.

[7] J. H. Chen and M. P. C. Fossorier, "Near optimum universal belief propagation based decoding of low-density parity check codes," *IEEE Transactions on Communications*, vol. 50, pp. 406-414, 2002.

[8] H. Jin, A. Khandekar, and R. McEliece, "Irregular repeat-accumulate codes," In. *Proc. 2nd International Symposium on Turbo Codes & Related Topics*, Brest, France, Sept 2000.

[9] M. Eroz, F. W. Sun, and L. N. Lee, "DVB-S2 low density parity check codes with near Shannon limit performance," *International Journal of Satellite Communications and Networking*, vol. 22, pp. 269-279, 2004.

[10] F. Kienle, T. Brack, and N. Wehn, "A Synthesizable IP Core for DVB-S2 LDPC Code Decoding," In. *Proc. Design, Automation and Test in Europe (DATE'05)*, Munich, Germany, Mar. 2005.

[11] J. Dielissen, A. Hekstra, and V. Berg, "Low cost LDPC decoder for DVB-S2," In. *Proc. Design, automation and test in Europe: Designers' forum (DATE'06)*, Munich, Germany, Mar. 2006.

[12] M. Gomes, G. Falcão, J. Gonçalves, V. Silva, M. Falcão, and P. Faia, "HDL Library of Processing Units for Generic and DVB-S2 LDPC Decoding," In. *Proc. International Conference on Signal Processing and Multimédia Applications (SIGMAP2006)*, Setúbal, Portugal, Aug. 2006.

[13] J. T. Zhang and M. P. C. Fossorier, "Shuffled iterative decoding," *IEEE Transactions on Communications*, vol. 53, pp. 209-213, 2005.

[14] H. Xiao and A. H. Banihashemi, "Graph-based message-passing schedules for decoding LDPC codes," *IEEE Transactions on Communications*, vol. 52, pp. 2098-2105, 2004.

[15] E. Sharon, S. Litsyn, and J. Goldberger, "An efficient message-passing schedule for LDPC decoding," *Electrical and Electronics Engineers in Israel, 2004. Proceedings. 2004 23rd IEEE Convention of*, pp. 223-226, 2004.