Available online at www.sciencedirect.com

ScienceDirect

Chaos, Solitons and Fractals 36 (2008) 500–511

CHAOS
SOLITONS & FRACTALS

www.elsevier.com/locate/chaos

Study of bifurcation behavior of two-dimensional turbo product code decoders

Yejun He, Francis C.M. Lau *, Chi K. Tse

Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hunghom, Hong Kong

Accepted 30 June 2006

Communicated by Prof M.S. El Naschie

Abstract

Turbo codes, low-density parity-check (LDPC) codes and turbo product codes (TPCs) are high performance error-correction codes which employ iterative algorithms for decoding. Under different conditions, the behaviors of the decoders are different. While the nonlinear dynamical behaviors of turbo code decoders and LDPC decoders have been reported in the literature, the dynamical behavior of TPC decoders is relatively unexplored. In this paper, we investigate the behavior of the iterative algorithm of a two-dimensional TPC decoder when the input signal-to-noise ratio (SNR) varies. The quantity to be measured is the mean square value of the posterior probabilities of the information bits. Unlike turbo decoders or LDPC decoders, TPC decoders do not produce a clear "waterfall region". This is mainly because the TPC decoding algorithm does not converge to "indecisive" fixed points even at very low SNR values.

1. Introduction

In the past decade, there has been a lot of interest in studying channel codes, e.g., turbo codes, low-density parity-check (LDPC) codes and turbo product codes (TPCs) [1–7]. The main motivation for studying these codes is that theoretically they can achieve performance very close (as close as 0.0045 dB) to the Shannon's limit [8]. Practically, such codes have also found applications in areas like mobile communications, optical communications and satellite communications [9–13]. Turbo codes and LDPC codes have relatively better performance compared to TPCs, but TPCs are becoming very competitive as fast parallel decoders can be implemented easily. While turbo codes, LDPC codes and TPCs are based on different encoding algorithms, their decoding methods share the same philosophy. All the decoders make use of nonlinear iterative algorithms to recover the codewords.

Due to their nonlinearity, the iterative algorithms exhibit phenomena such as fold, flip and Neimark-Sacker bifurcations. In particular, it has been reported that turbo decoders and LDPC decoders converge to indecisive fixed

* Corresponding author.
E-mail address: encmlau@polyu.edu.hk (F.C.M. Lau).

0960-0779/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.chaos.2006.06.101

points with errors occurring at the low signal-to-noise (SNR) region, and to unequivocal fixed points with all codewords decoded correctly at the high SNR region [1,5,14]. Moreover, from the low SNR region to the high SNR region, there is a transition region known as the "waterfall region" where periodic and chaotic behaviors are observed in the decoder. Feedback mechanisms have also been proposed to enhance the decoder performance in the waterfall region [14,15].

In this paper, we focus on the dynamical behavior of the two-dimensional TPC decoders under an additive white Gaussian noise channel. We observe that unlike turbo code and LDPC decoders, TPC decoders do not converge to indecisive fixed points even at very low SNR values. Rather, they produce periodic orbits and chaos. Because of this, there is no well-defined "waterfall region". Nonetheless, when the SNR value is large enough, the TPC decoders converge to unequivocal fixed points. We organize this paper as follows. In Section 2, we briefly review the operation principles of the TPC encoder and decoder. In Section 3, we present the measure by which the dynamical behavior of the TPC decoders is traced. Detailed simulation results and discussions are then given. Conclusions will be drawn in Section 4.

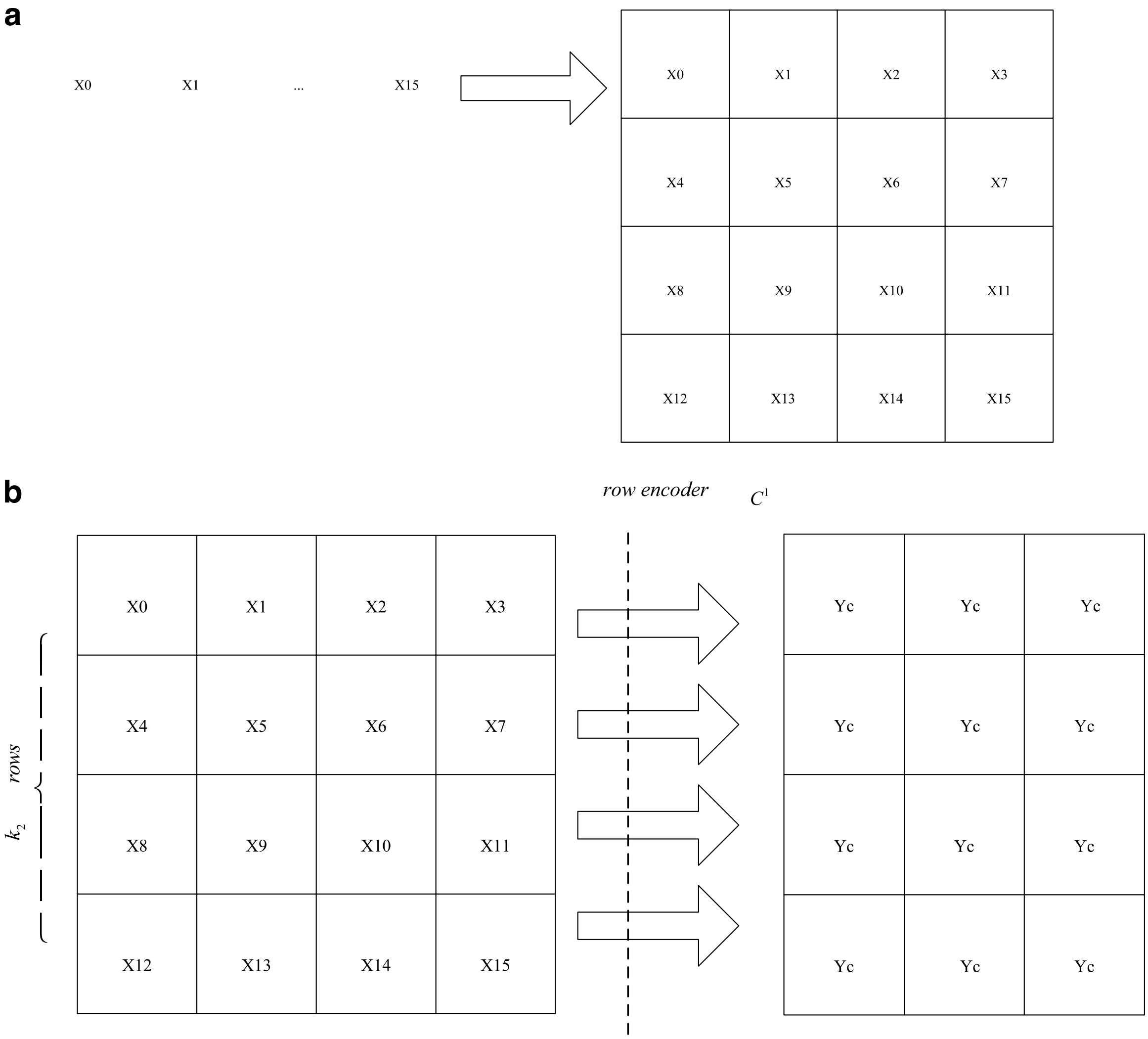

Fig. 1. Encoding procedures of a 2-D turbo product code. (a) Information bits form the entries of a 4 × 4 matrix, (b) row encoding and (c) column encoding.

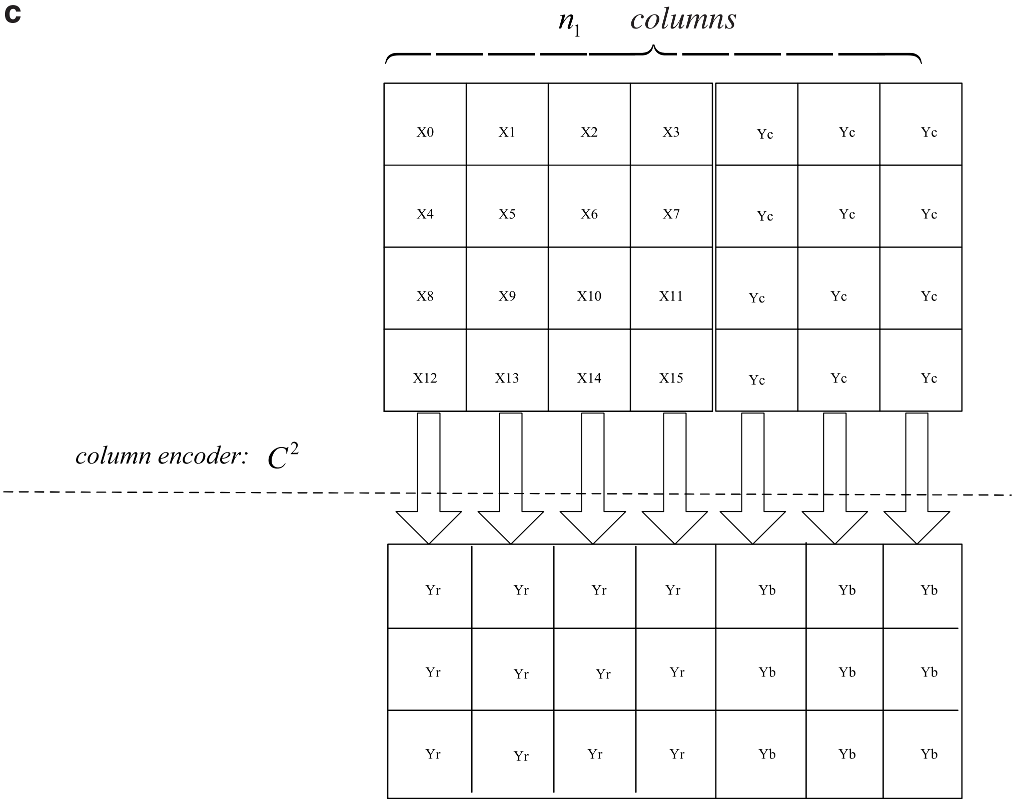

Fig. 1 (*continued*).

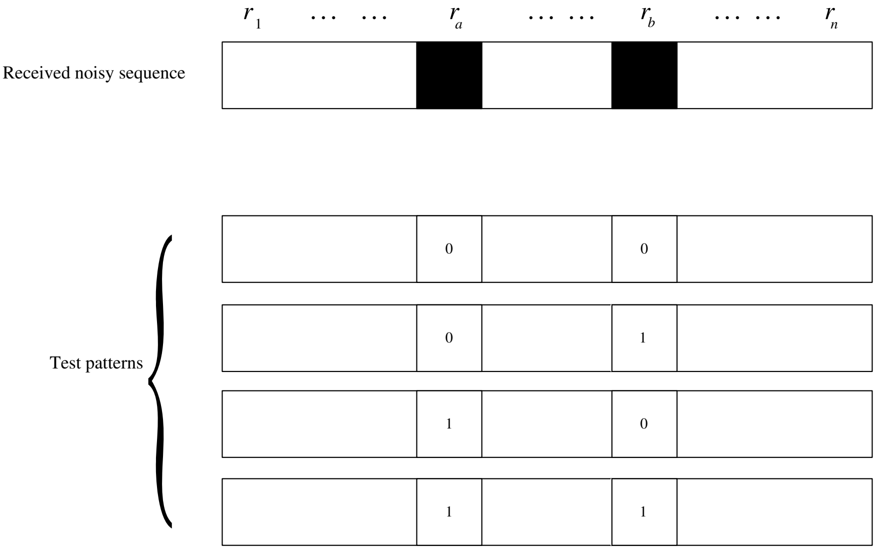

Fig. 2. Test patterns generated by the Chase algorithm, $p = 2$.

2. Review of turbo product codes

2.1. Encoding of TPC codewords

Turbo product codes (TPC), also known as block turbo codes, are built from two-, three-, or multi-dimensional array of block codes. In general, the larger the dimension, the higher the TPC complexity. Also, the constituent codes

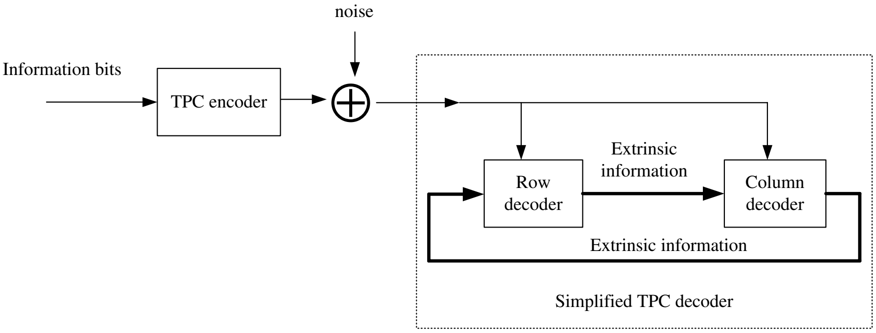

Fig. 3. Iterative process of a 2-D TPC decoder.

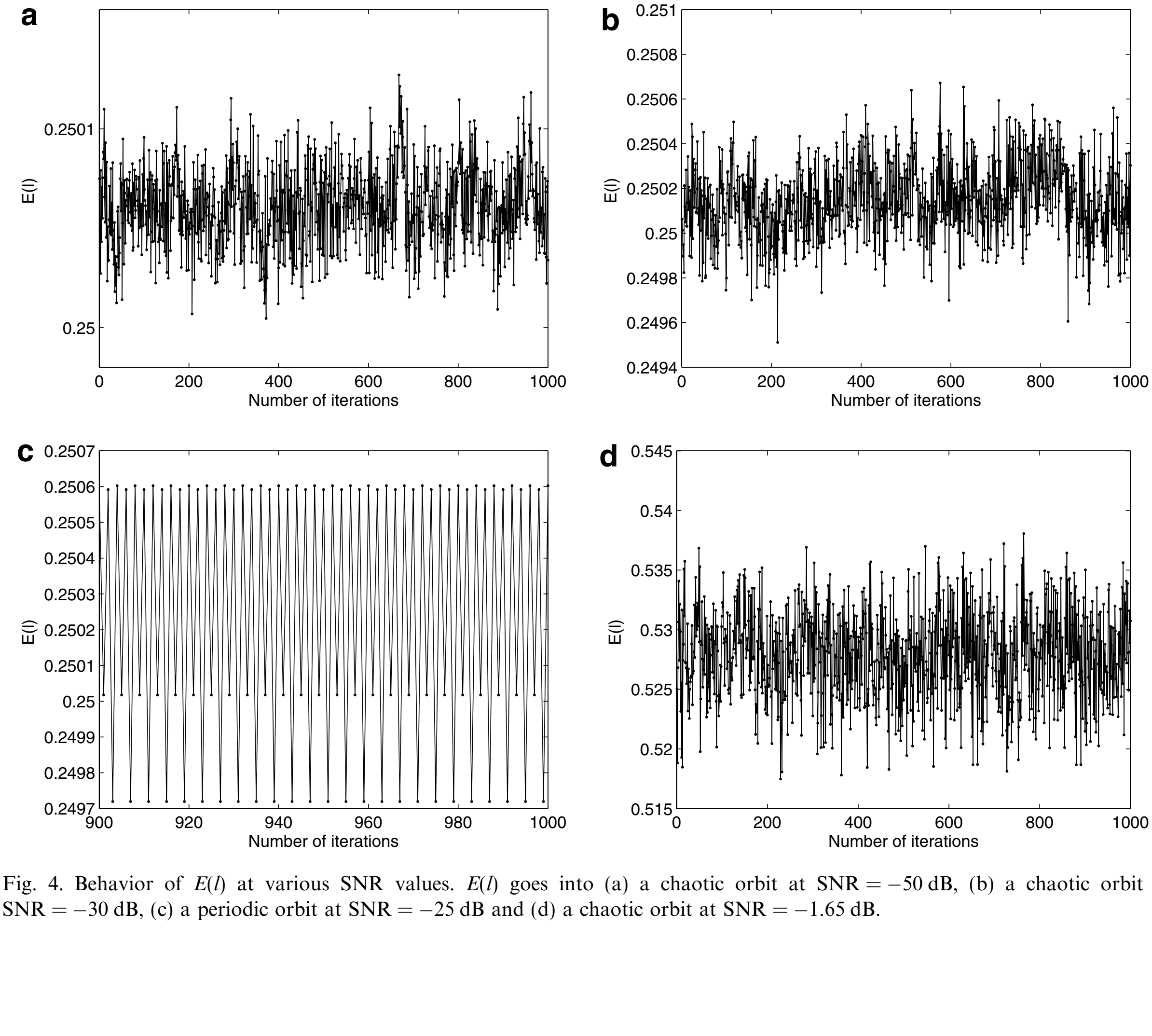

Fig. 4. Behavior of $E(l)$ at various SNR values. $E(l)$ goes into (a) a chaotic orbit at SNR $= -50$ dB, (b) a chaotic orbit at SNR $= -30$ dB, (c) a periodic orbit at SNR $= -25$ dB and (d) a chaotic orbit at SNR $= -1.65$ dB.

forming TPC can be different, though they are usually identical for simple implementation. Suppose the constituent codes are of the same type. Then, dependent upon the type of constituent codes, TPCs can be categorized into Hamming product codes, extended Hamming product codes, single-parity-check product codes, etc.

Since high dimensional TPCs are very complicated and are not easy to implement, we focus our study on 2-D TPCs. For example, given the Hamming codes $C^i = (n_i, k_i, \delta_i)$; $i = 1, 2$, where n_i, k_i and δ_i denote, respectively, the codeword length, the number of information bits and the minimum Hamming distance of the ith Hamming code. A 2-D TPC, denoted by $(n, k, d_{\min}) = (n_1 n_2, k_1 k_2,\ d_{\min})$, can be obtained by arranging the information bits in a rectangular array, and encoding the rows and then columns with the row encoder and the column encoder, respectively. It has also been shown that in the resultant coded matrix, all rows are codewords of C^1 and all columns are codewords of C^2. Based on this property, it can be proved that the minimum distance of the TPC, denoted by $d_{\min}$, is equal to the product of the minimum distances of the constituent codes, i.e., $d_{\min} = \delta_1 \delta_2$.

In Fig. 1, we illustrate the steps for encoding a 2-D TPC in which the constituent codes are both the (7,4) Hamming code. First, the information bits form the entries of a 4 × 4 matrix, as shown in Fig. 1(a). Then, each row of elements is encoded by the $C^1 = (7, 4)$ Hamming code, which adds $7 - 4 = 3$ check bits to the end of each row, as in Fig. 1(b). Finally, in Fig. 1(c), each of the columns is further encoded by $C^2 = (7, 4)$ Hamming code, appending 3 other check bits to the end of each column. Thus, we obtain a TPC with parameters $n = 49$, $k = 16$ and $d_{\min} = 9$. Since the TPC encoder produces check bits on check bits, the final codeword is different from the parallel concatenated product code which does not contain any check bits on check bits [16]. Also, in the aforementioned encoding procedures, the row encoding and the column encoding steps can be interchanged, i.e., column encoding can be performed prior to row encoding, without affecting the output codewords.

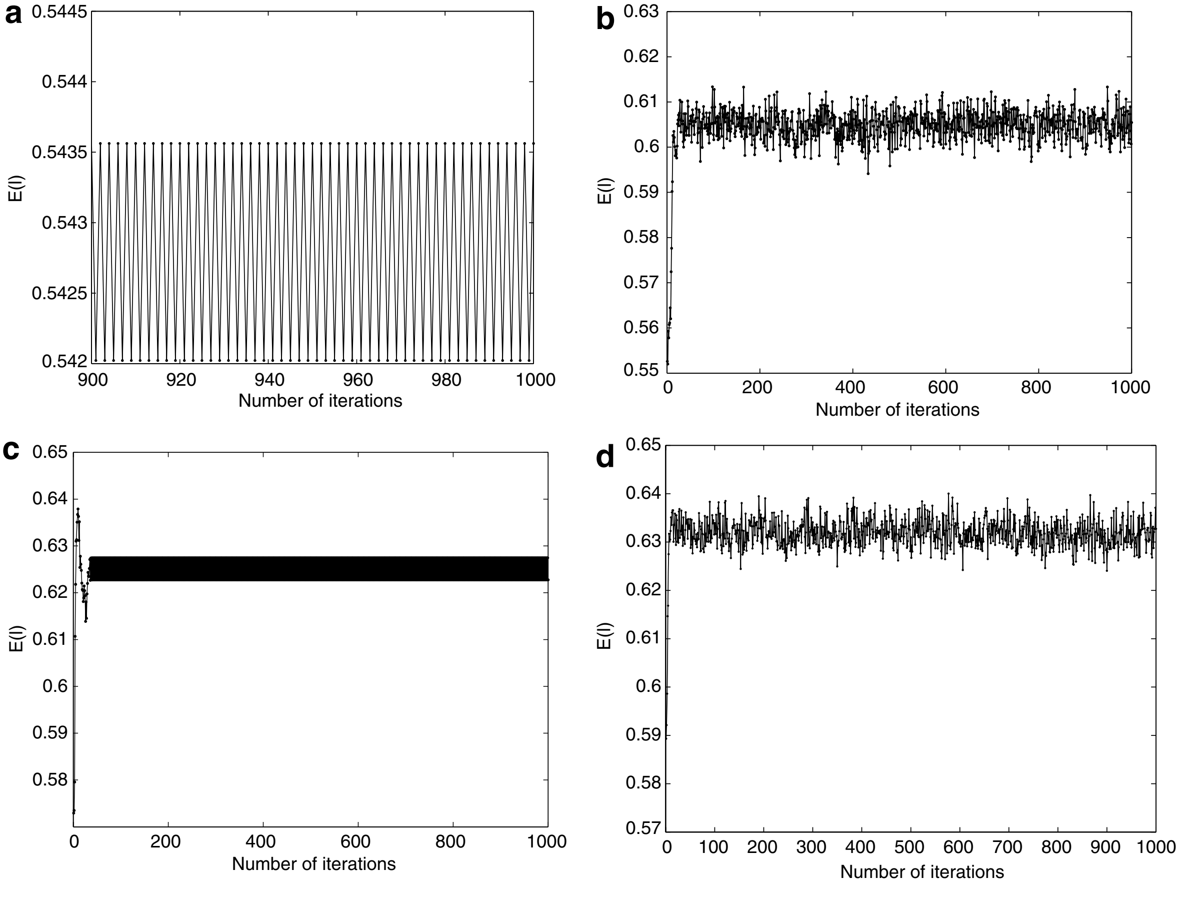

Fig. 5. Behavior of $E(l)$ at various SNR values. $E(l)$ goes into (a) a periodic orbit at SNR = −1.45 dB, (b) a chaotic orbit at SNR = −1.3 dB, (c) a periodic orbit at SNR = −1.1 dB and (d) a chaotic orbit at SNR = −0.9 dB.

2.2. Decoding of TPC codewords

2.2.1. Decoding of block codes

We first consider the decoding of a block code using the well-known decoding technique, the Chase algorithm [17]. Denote the transmitted codeword by

$$\boldsymbol{c} = [\,c_1 \quad c_2 \quad \cdots \quad c_n\,] \tag{1}$$

where $c_i \in \{0, 1\}$. Then, we map the components of the codeword as 0 → −1 and 1 → +1. The transmitted signal vector is presented by

$$\boldsymbol{s} = [\,s_1 \quad s_2 \quad \cdots \quad s_n\,] = [\,2c_1 - 1 \quad 2c_2 - 1 \quad \cdots \quad 2c_n - 1\,] \tag{2}$$

where $s_i \in \{-1, 1\}$. Assuming the channel contains additive white Gaussian noise (AWGN) only, the received noisy signal vector, denoted by $\boldsymbol{r}$, is given by

$$\boldsymbol{r} = [\,r_1 \quad r_2 \quad \cdots \quad r_n\,] = \boldsymbol{s} + \boldsymbol{\eta} \tag{3}$$

where the vector $\boldsymbol{\eta} = [\,\eta_1 \quad \eta_2 \quad \cdots \quad \eta_n\,]$ consists of noisy elements with zero mean and variance σ^2. The signal-to-noise ratio (SNR) of the system is thus equal to $1/\sigma^2$.

After receiving the noisy codeword $\boldsymbol{r}$ from the channel, we first determine the p least reliable bit positions if hard decisions are to be made on all bits, where p is usually a small number. Then, we replace the noisy signals at these bit positions by all possible bit combinations and form 2^p test patterns. In Fig. 2, four test patterns are created based on the received noisy codeword. Note that except in the p bit positions, the received signals corresponding to other bit

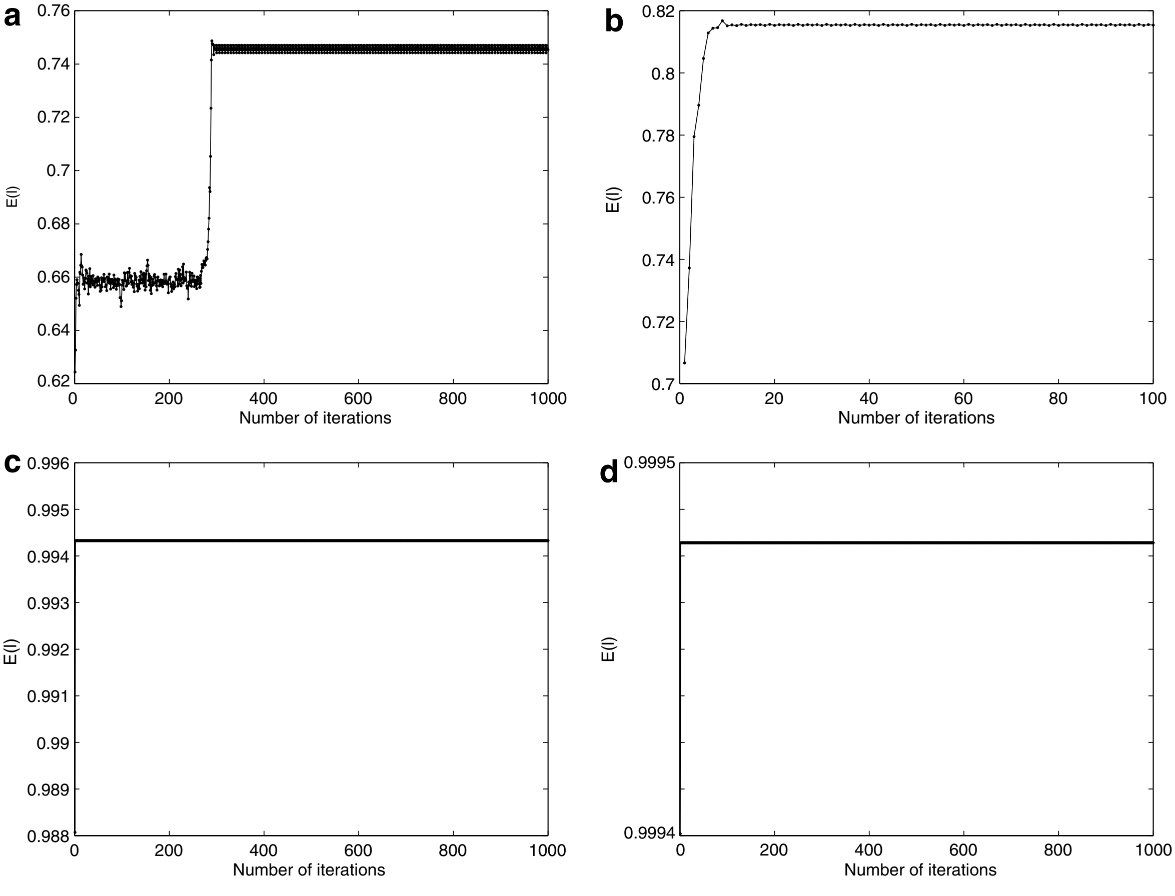

Fig. 6. Behavior of $E(l)$ at various SNR values. $E(l)$ goes into (a) a periodic orbit at SNR = −0.5 dB, (b) a periodic orbit at SNR = 0.5 dB, (c) an unequivocal fixed point at SNR = 4.5 dB and (d) an unequivocal fixed point at SNR = 6 dB.

positions remain unchanged. After passing the test patterns to a hard-decision decoder, 2^p possible codewords, denoted by $\boldsymbol{d}^{(i)}$, $i = 1, 2, \ldots, 2^p$, are obtained, among which the most likely one will be chosen as the decoded codeword according to the following rule:

$$\boldsymbol{d} = [\,d_1 \quad d_2 \quad \cdots \quad d_n\,] = \boldsymbol{d}^{(j)} \quad \text{s.t. } |\boldsymbol{r} - (2\boldsymbol{d}^{(j)} - \mathbf{1})| \leqslant |\boldsymbol{r} - (2\boldsymbol{d}^{(i)} - \mathbf{1})| \quad \forall i \in \{1, 2, \ldots, 2^p\} \tag{4}$$

where

- $d_m \in \{0, 1\}$; $m = 1, 2, \ldots, n$;
- $\mathbf{1}$ denotes the all-ones vector; and
- $|\boldsymbol{x} - \boldsymbol{y}|$ represents the Euclidean distance between $\boldsymbol{x}$ and $\boldsymbol{y}$, i.e.,

$$|\boldsymbol{x} - \boldsymbol{y}| = \sqrt{\sum_{m=1}^{n} (x_m - y_m)^2}. \tag{5}$$

2.2.2. Decoding of TPCs

The usual Chase algorithm is not applied directly to TPC decoding because the sub-decoders of TPC should produce soft outputs (extrinsic information) during the iterative process. The extended Chase algorithm [6], which produces soft outputs, is used instead in the TPC decoder. A typical TPC decoder consists of a row sub-decoder and a column sub-decoder which work iteratively one after the other, as shown in Fig. 3. Denote the signal matrix (size $n \times n$) received via the AWGN channel by $\boldsymbol{R}$. Also, denote the soft input matrices to the row sub-decoder and column sub-decoder by $\boldsymbol{R}_{\text{row}}$

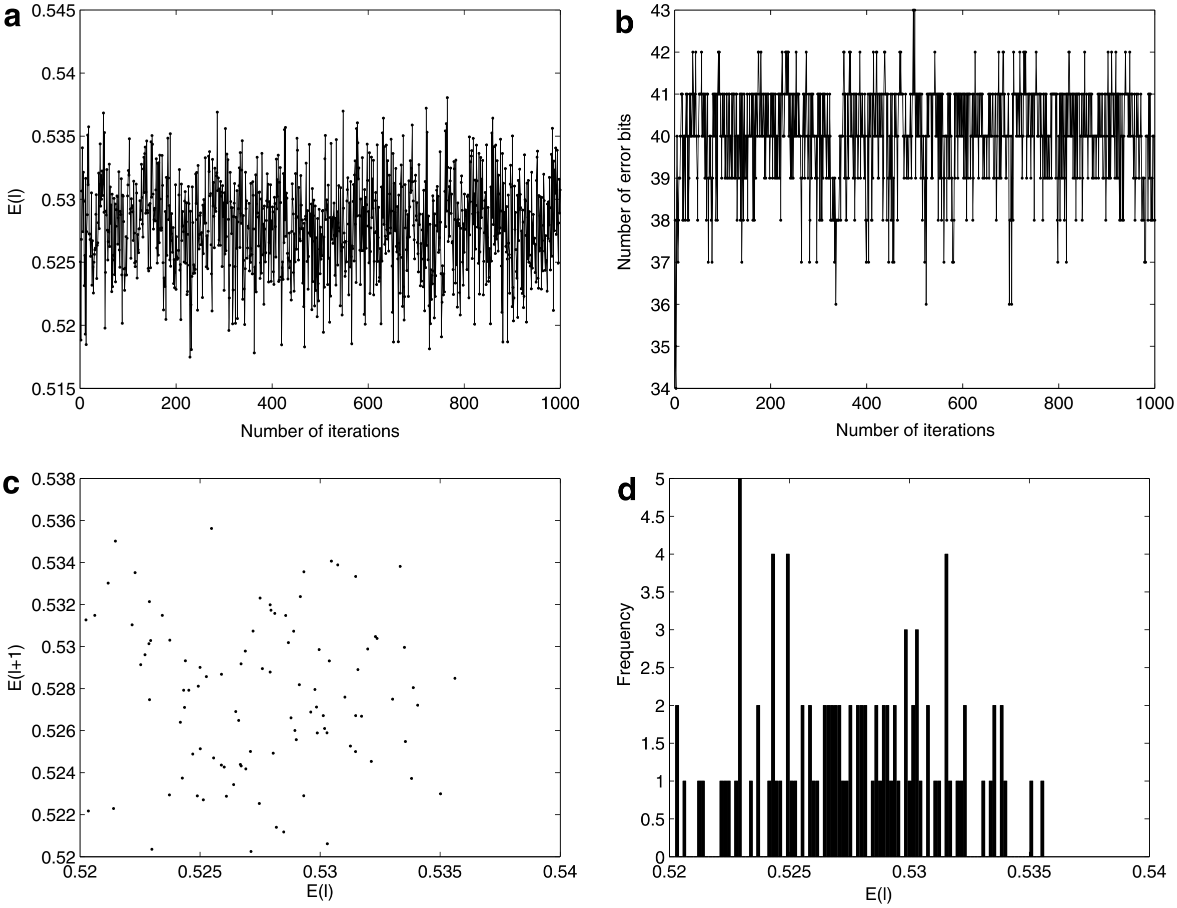

Fig. 7. Chaotic behavior of turbo product code decoder at SNR = −1.65 dB. (a) $E(l)$ versus number of iterations, (b) number of information error bits versus number of iterations, (c) $E(l+1)$ versus $E(l)$ and (d) histogram of $E(l)$ for the last 100 iterations.

and $\boldsymbol{R}_{\text{col}}$, respectively. Note that the soft input matrices change as the iterative process goes on and each entry in the matrices corresponds to a coded bit received.

In the first iteration of the row sub-decoder, the soft input matrix equals the received signal matrix, i.e., $\boldsymbol{R}_{\text{row}} = \boldsymbol{R}$. Consider any one of the rows in $\boldsymbol{R}_{\text{row}}$. In the extended Chase algorithm, two soft outputs are generated for each bit by the sub-decoder. One of the outputs corresponding to the mth bit of the row under consideration is given by [6]

$$r'_m = \begin{cases} \left(|\boldsymbol{r} - (2\boldsymbol{d} - \boldsymbol{1})|^2 - |\boldsymbol{r} - (2\boldsymbol{d}^{(q)} - \boldsymbol{1})|^2\right)\frac{2d_m - 1}{4} & \text{if } \boldsymbol{d}^{(q)} \text{ exists} \\ \beta(2d_m - 1) & \text{if } \boldsymbol{d}^{(q)} \text{ does not exist} \end{cases} \tag{6}$$

where

- $\boldsymbol{r}$ is the row vector;
- $\boldsymbol{d}$ is the decoded codeword found using (4);
- $\boldsymbol{d}^{(q)}$: $q \in \{1, 2, \ldots, 2^p\}$ represents the possible codeword satisfying $d_m \neq d_m^{(q)}$ and $(2\boldsymbol{d}^{(q)} - \boldsymbol{1})$ is at a minimum Euclidean distance from $\boldsymbol{r}$ among the possible codewords; and
- β is the reliability constant.

Another soft output, denoted by w_m, is found by subtracting r_m from r'_m, i.e.,

$$w_m = r'_m - r_m \tag{7}$$

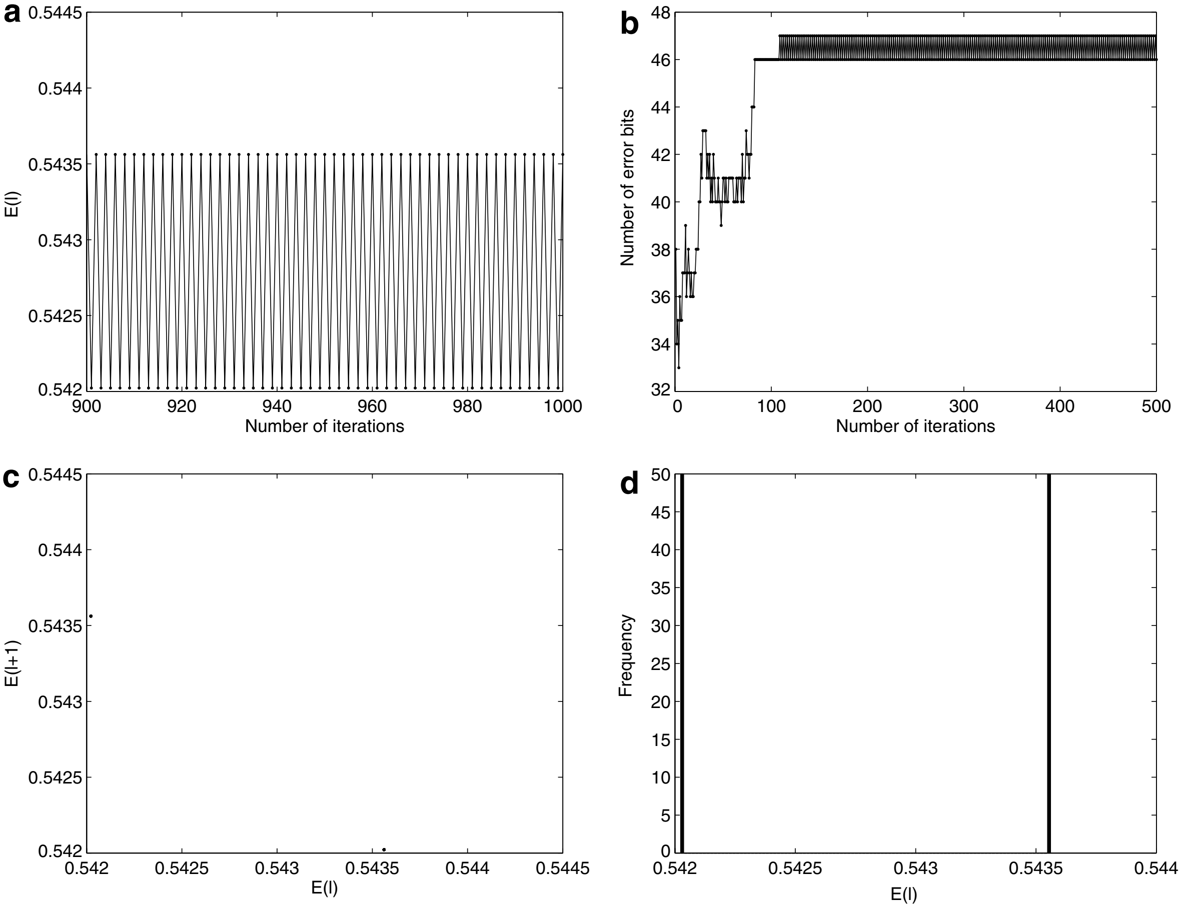

Fig. 8. Periodic behavior of turbo product code decoder at SNR = −1.45 dB. (a) $E(l)$ versus number of iterations, (b) number of information error bits versus number of iterations, (c) $E(l+1)$ versus $E(l)$ and (d) histogram of $E(l)$ for the last 100 iterations.

The above algorithm is applied to all rows of the soft input matrix and a soft output matrix, denoted by $\boldsymbol{W}_{\text{row}}$ and in which elements are those found using (7), is produced. As for the soft input matrix for the column sub-decoder, it is computed from

$$\boldsymbol{R}_{\text{col}} = \boldsymbol{R} + \alpha \boldsymbol{W}_{\text{row}} \tag{8}$$

where α is a scaling factor. Based on the same principles in (6) and (7), the column sub-decoder generates soft outputs r'_m and w_m for the mth bit of each column vector. Similarly, a soft output matrix $\boldsymbol{W}_{\text{col}}$ is formed and the soft input matrix for the row sub-decoder is updated using

$$\boldsymbol{R}_{\text{row}} = \boldsymbol{R} + \alpha \boldsymbol{W}_{\text{col}}. \tag{9}$$

The iteration process continues until the maximum number of iterations has been reached or convergence has taken place.

3. Dynamics of 2-D TPC decoder

3.1. System parameters

As discussed in the previous section, there are several adjustable parameters in the TPC decoder. They are the test pattern parameter (p), the weighting factor (α) and the reliability factor (β). Each of them can alter the dynamics of the decoder. Here we set the values of p, α and β as 2, 0.5 and 1, respectively. These values are typically used in other TPC studies [6].

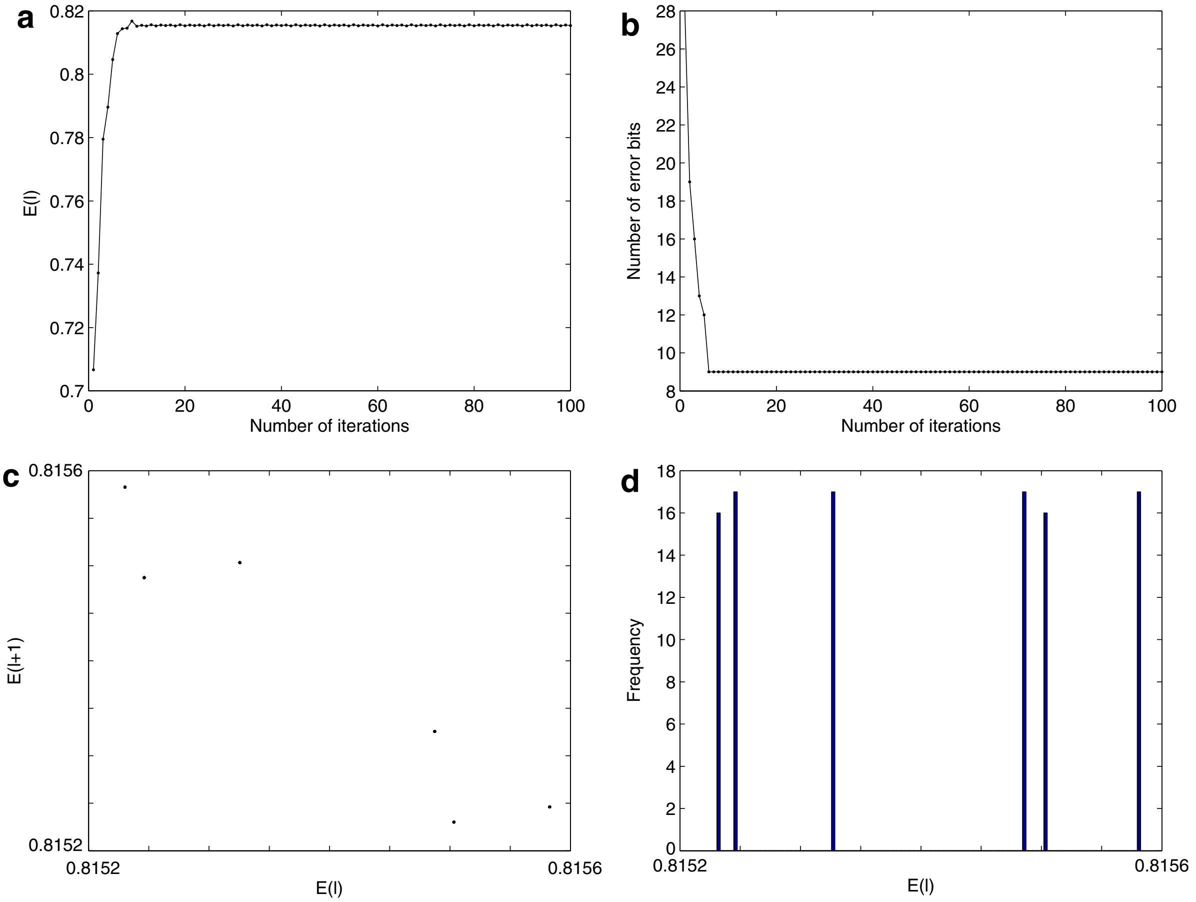

Fig. 9. Periodic behavior of turbo product code decoder at SNR=0.5 dB. (a) $E(l)$ versus number of iterations, (b) number of information error bits versus number of iterations, (c) $E(l+1)$ versus $E(l)$ and (d) histogram of $E(l)$ for the last 100 iterations.

Moreover, we assume that the ratios of consecutive noise samples, i.e., $\eta_1/\eta_2, \eta_2/\eta_3, \ldots, \eta_{n-1}/\eta_n$, are fixed and the all-zero codeword is being transmitted [5]. After making the aforementioned arrangements, we observe the dynamical behavior of the decoder as the signal-to-noise ratio (SNR) varies. To study the dynamics of the TPC decoder, we introduce a measure, denoted by E, that represents the mean-square value of the posterior probabilities of the decoded bits being equal to zero and is approximated by [6]

$$E = \frac{1}{n^2}\sum_m \Pr{}^2[m\text{th information bit} = 0] \approx \frac{1}{n^2}\sum_m \frac{1}{\left[1+\exp\left(\frac{2}{\sigma^2}\times r'_{m,d_m=0}\right)\right]^2}. \tag{10}$$

Note that in (10), $r'_{m,d_m=0}$ is always less than zero. Therefore, when the SNR increases (σ^2 decreases), the exponential function tends to zero and the measure E converges to unity. Here, the measure is computed based on the soft outputs of the column sub-decoder.

We select the TPC $(15,11)^2$ and a random noise realization. We then perform simulations with SNR ranging from −50 dB to 6 dB. For each SNR value, we perform 1000 full iterations (consisting of one row sub-decoding and one column sub-decoding) and record the changes of the signals of the decoder.

3.2. Simulation results

To study the changing dynamical behavior of the TPC decoder, we plot the trajectory of $E(l)$ against l, where l denotes the iteration number. Figs. 4–6 show the results as the SNR value is increased. Starting at SNR = −50 dB

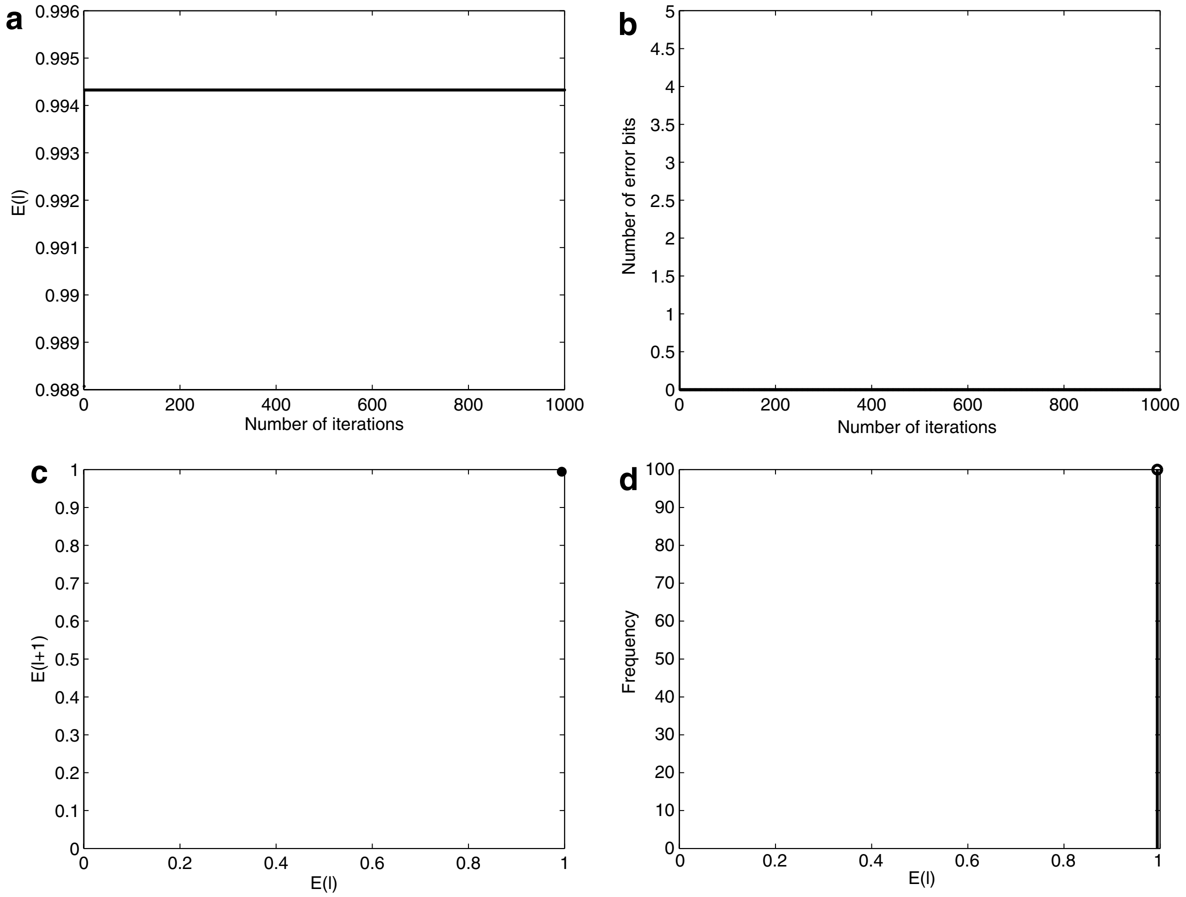

Fig. 10. Turbo product code decoder converges to an unequivocal fixed point at SNR=4.5 dB. (a) $E(l)$ versus number of iterations, (b) number of information error bits versus number of iterations, (c) $E(l+1)$ versus $E(l)$ and (d) histogram of $E(l)$ for the last 100 iterations.

at which the phase trajectory of $E(l)$ is chaotic, as shown in Fig. 4(a), we gradually increase the SNR. We observe that the first bifurcation occurs between SNR values of −30 dB and −25 dB. Fig. 4(b) and (c) shows that the phase trajectory of $E(l)$ changes from chaotic to periodic when the SNR is increased from −30 dB to −25 dB. When the SNR is increased further, we find that the periodic points lose their stability and bifurcate at around SNR = −1.7 dB. Fig. 4(d) shows a chaotic trajectory at SNR = −1.65 dB. Further increasing the SNR creates further bifurcations. Fig. 5(a) and (b) depicts the periodic trajectory at SNR = −1.45 dB and the chaotic trajectory at SNR = −1.3 dB, respectively. When we increase the SNR again, periodic trajectories and chaotic trajectories occur in subsequent SNR regions. In Fig. 5(c) and (d), the trajectories of $E(l)$ are shown at SNR values of −1.1 dB and −0.9 dB. The trajectory converges to a periodic orbit again when SNR reaches −0.5 dB. Fig. 6(a) and (b) plots the trajectories at SNR values of −0.5 dB and 0.5 dB, respectively. Finally, when the SNR value is large enough, the TPC decoder arrives at an unequivocal fixed point after a number of iterations. The trajectories corresponding to SNR = 4.5 dB and 6 dB are shown in Fig. 6(c) and (d), respectively.

Figs. 7–10 further show the typical behaviors of the TPC decoder in different SNR regions. In Fig. 7, at an SNR value of −1.65 dB, we observe that the TPC decoder behave chaotically. Throughout the 1000 iterations, the value of $E(l)$ changes within $[0.515, 0.54]$ without any fixed pattern. Also, the number of error bits changes erratically with the number of iterations. In Fig. 7(c), a plot of $E(l+1)$ versus $E(l)$ for the last 100 iterations illustrates that all the points are distinct. The histogram of these 100 values is then plotted in Fig. 7(d), which clearly shows that the value of $E(l)$ spreads over a wide range. In Fig. 8, at an SNR value of −1.45 dB, we observe that the TPC decoder converges to a periodic orbit after about 120 iterations. Fig. 8(c) also illustrates the period of the trajectory is 2 (note the two points near the value 0.5435 on both axes). Fig. 9 depicts the behavior of the TPC decoder at an SNR value of 0.5 dB. Similar to the case when SNR = −1.45 dB, the decoder produces a periodic orbit. The plot of $E(l+1)$ versus $E(l)$ at the steady state, as shown in Fig. 9(c), discloses that the orbit has a period of 6. As can be judged from Fig. 9(c), the number of error bits remains at 9 after the decoder has entered into the periodic orbit. Finally, with sufficiently large SNR, the decoder converges to an unequivocal fixed point. Fig. 10 shows such a case with SNR = 4.5 dB. It can be clearly seen that the decoder converges to the correct codeword very quickly.

4. Conclusion

In this paper, we have investigated the behavior of a two-dimensional turbo product code (TPC) decoder. Like turbo codes and low-density parity-check (LDPC) codes, TPC relies on the use of iterative algorithms for decoding the codewords. Unlike the turbo and LDPC decoding algorithms, however, the TPC decoding algorithm does not produce a clear “waterfall region”, below and above which the algorithm is expected to converge to indecisive fixed points and unequivocal fixed points, respectively. The main reason is that the TPC decoder does not converge to any “indecisive” fixed points. While bifurcations do occur at low signal-to-noise ratio (SNR) values, the TPC decoder trajectory mainly changes from periodic orbits to chaotic orbits, and vice versa. Convergence to indecisive points is not observed even at an SNR of −50 dB. Similar to the trajectories of the turbo decoders and LDPC decoders, the TPC decoder trajectory eventually converges to unequivocal fixed points when the SNR value is large enough.

Acknowledgement

This work was supported in part by a Hong Kong Polytechnic University research grant (A-PG57).

References

[1] Richardson T. The geometry of turbo-decoding dynamics. IEEE Trans Inform Theory 2000;46(1):9–23.

[2] Agrawal D, Vardy A. The turbo decoding algorithm and its trajectories. IEEE Trans Inform Theory 2001;47(2):699–722.

[3] Duan L, Rimoldi B. The iterative turbo decoding algorithm has fixed points. IEEE Trans Inform Theory 2001;47(7):2993–5.

[4] Chen JH, Fossorier MPC. Near optimum universal belief propagation based decoding of low-density parity check codes. IEEE Trans Commun 2002;50(3):406–14.

[5] Zheng X, Lau FCM, Tse CK, Wong SC. Study of bifurcation and chaos of LDPC decoders. In: European conference on circuit theory and design 2005, paper 207.

[6] Pyndiah R. Near optimum decoding of product codes: block turbo codes. IEEE Trans Commun 1998;46(8):1003–10.

[7] Hirst SA, Honary B, Markarian G. Fast Chase algorithm with an application in turbo decoding. IEEE Trans Commun 2001;49(10):1693–9.

[8] Chung SY, Forney JGD, Richardson T, Urbanke R. On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit. IEEE Commun Lett 2001;5:58–60.

[9] Fonseka JP. Application of turbo codes in satellite mobile systems. Electron Lett 1999;35(2):114–5.

[10] Shibutani A, Suda H, Adachi F. Multistage recursive interleaver for turbo codes in DS-CDMA mobile radio. IEEE Trans Veh Tech 2002;51(1):88–100.

[11] Vasic B, Djordjevic IB. Low-density parity check codes for long haul optical communications systems. IEEE Photon Technol Lett 2002;14:1208–10.

[12] Mizuochi T, Miyata Y, Kobayashi T, Ouchi K, Kuno K, Kubo K, et al. Forward error correction based on block turbo code with 3-bit soft decision for 10-Gb/s optical communication systems. IEEE J Sel Top Quantum Electron 2004;10(2):376–86.

[13] Janakiraman V, Jang JS, Kim J. Transmission of block turbo coded digital audio over FM-SCA. IEEE Trans Broadcasting 2004;50(2):07–112.

[14] Kocarev L, Tasev Z, Vardy A. Improving turbo codes by control of transient chaos in turbo-decoding algorithms. Electron Lett 2002;38(20):1184–6.

[15] Zheng X, Lau FCM, Tse CK, Wong SC. Techniques for improving block error rate of LDPC decoders. IEEE Int Sym Circ Sys 2006:2261–4.

[16] Sella A, Beery Y. Convergence analysis of turbo decoding of product codes. IEEE Trans Inform Theory 2001;47(2):723–35.

[17] Chase D. A class of algorithms for decoding block codes with channel measurement information. IEEE Trans Inform Theory 1972;18(1):170–82.