Turbo Product Codes: Applications, Challenges, and Future Directions

H. Mukhtar, *Member, IEEE*, A. Al-Dweik, *Senior Member, IEEE*, and A. Shami, *Senior Member, IEEE*

***Abstract*—Turbo product codes (TPCs) have been integrated in several practical applications, and hence, they have been considered widely in the literature where the main aim is improving the error performance and/or reducing the computational and implementation complexity. This paper presents a comprehensive survey of the research that focuses on TPCs in terms of encoding, decoding, error performance, and complexity. Moreover, this paper also considers the advantages of integrating TPCs in hybrid automatic repeat request systems where power optimization becomes very efficient and the complexity can be reduced using the unique properties of TPCs such as error self-detection capabilities. Based on the surveyed literature, the pivotal open research issues in TPCs are presented and discussed.**

***Index Terms*—Turbo codes, product codes, error control coding, error correction, iterative decoding, error detection, automatic repeat request, soft decision decoding, energy efficiency, complexity reduction.**

I. Introduction

FORWARD error correction (FEC) is one of the key tools that enabled the explosive growth of the wireless communications industry in the last decade. The basic role of FEC is to provide the users with reliable digital transmission using minimum excess power, bandwidth and complexity. Consequently, FEC substantially contributed to the success of transformation towards the A^5 vision (anyone to access anything from anywhere at anytime on any device). Such vision has placed stringent quality of service (QoS) requirements, which require optimal design at all layers of the wireless communications protocol stack to reduce the cost, power consumption and complexity while increasing the capacity, coverage and reliability. Watching high definition television (HDTV) on small-size mobile devices is an example for such extreme QoS requirements because such application requires up to 34 Mbps with packet error rate less than 10^{-6} [1]–[3]. The problem becomes even more challenging when such mobile devices are used for transmission. For example, video conferencing requires up to 1.92 Mbps and packet error rate of less than 10^{-4} [2]. In this survey, we consider turbo product codes (TPCs) since they are one of the primary FEC techniques where we summarize the major contributions, present state-of-the-art results, and present the main advantages and disadvantages of TPCs as compared to other well established FEC techniques.

Manuscript received February 18, 2016; revised May 26, 2016; accepted June 27, 2016. Date of publication July 7, 2016; date of current version November 18, 2016.

H. Mukhtar is with the Department of Electrical and Computer Engineering, Khalifa University, Abu Dhabi 27788, UAE (e-mail: husameldin.mukhtar@kustar.ac.ae).

A. Al-Dweik is with the Department of Electrical and Computer Engineering, Khalifa University, Abu Dhabi 27788, UAE, and also with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada (e-mail: dweik@kustar.ac.ae).

A. Shami is with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada (e-mail: ashami@eng.uwo.ca).

Digital Object Identifier 10.1109/COMST.2016.2587863

A. Overview of TPCs

TPCs, alternatively referred to as block turbo codes (BTC), are powerful FEC codes that can be implemented with reasonable complexity [4]. They support a wide range of codeword sizes and code rates. Product codes are first introduced by Elias [5]. Similar to turbo codes [6], product codes are constructed using an inner and outer code separated by an interleaver. Classical product codes are obtained from two linear block codes applied serially on the two dimensions of a matrix. Moreover, product codes have become popular after the introduction of a reasonable complexity soft-input soft-output (SISO) iterative decoding algorithm in [7].

Compared to other capacity-approaching codes, TPCs have several advantages such as simple encoding/decoding and high coding gain at high code rates [8]–[10]. Moreover, TPCs are highly parallelizable which makes them suitable for high speed applications [11]–[15].

TPCs have large minimum Hamming distances; hence, they do not suffer from error floors as much as the original class of turbo codes known as parallel concatenation convolutional codes (PCCCs) [6], [16]. Moreover, TPCs can support high code rates using high rate component codes. Such approach is different from PCCCs where a large amount of puncturing is required to produce high code rates, which may degrade the error performance and increase decoding complexity [17]. Similar to the error performance and complexity, code latency of various coding techniques has attracted noticeable research attention [18]–[21]. The comparison between different coding schemes is typically performed based on the assumption of infinite processing speed and hence the latency is solely determined by the codeword length [18]–[21]. In such scenarios, PCCCs will have lower latencies than TPCs. However, TPCs can be designed to have low codeword lengths, and hence, high latencies can be avoided. Moreover, TPCs have inherent error detection capability which can be used to terminate the iterative decoding process without the need for additional parity check bits [22]. Therefore, TPCs can be considered as an attractive solution for high speed communication systems with very low bit error rate (BER) requirements [23]–[27].

Based on the assumption that latency is solely determined by the codeword length [18]–[21], LDPC codes and TPCs can have equivalent latencies under equal BER constraints and high codeword lengths, however, TPCs outperform LDPC codes at low codeword lengths. Some numerical examples are given in Section III-E.

B. TPCs Applications

TPCs are currently included in various communication standards such as the IEEE 802.16 for fixed and mobile broadband wireless access systems [28], which is commercially known as the Worldwide Interoperability for Microwave Access (WiMAX), IEEE 802.20 Mobile Broadband Wireless Access (MBWA) for local and metropolitan area networks [29], and IEEE-1901 for broadband power line networks [30]. Moreover, TPCs have been proposed for many applications such as optical communications, satellite systems, multimedia transmission and data storage devices.

The high coding gain of TPCs at high code rates has been utilized in optical communication systems to achieve reliable high speed transmission and improved system capacity. In [9], TPCs with 20% overhead are employed in an optical code division multiple access (OCDMA) system to reduce the weight and length of active users' optical orthogonal codes achieving a 50% bandwidth reduction. The effectiveness of TPCs in high bit rate optical transmission is experimentally demonstrated in [23] for a 10-Gb/s system. TPCs have also been proposed for optical communication systems in [31] and [32] where TPCs decoders are optimized for practical optical channel models.

Moreover, TPCs are employed to improve the performance of satellite communication systems. In [33], a TPC hardware implementation is used to demonstrate the spectral efficiency improvement when TPCs are used in a digital video broadcasting satellite (DVB-S) system. TPCs are also used in [34] as an unequal error protection (UEP) scheme for satellite communications where packet headers are given more importance than payload. A commercial broadband satellite system known as IPSTAR employs TPCs in its communication terminals to provide high-throughput satellite services [35], [36].

TPCs are proposed for many image, video and audio applications to enhance transmission efficiency [37]–[39]. Multimedia content has unequal importance and the loss of some source information has higher distortion impact on the received quality compared to other less important source data. Therefore, for multimedia applications such as image and video transmission, TPCs are usually designed to provide different levels of protection based on the importance of source information [40]–[44].

TPCs and low density parity check (LDPC) codes are of great interest in data storage applications which have strict requirements for low decoding errors and high data rate [10]. Nevertheless, TPCs have simple encoding and decoding schemes and better error statistics for magnetic recording channels [8]. A practical FEC solution is proposed in [45] for magnetic recording systems where an LDPC code is concatenated with a single parity check (SPC) code to construct a product code. The complexity of the constructed product code and regular LDPC code are comparable, but the product code offers significant signal-to-noise ratio (SNR) gain.

This article presents a comprehensive survey of TPCs where we summarize the main contributions reported in the literature in terms of encoding, decoding and error performance enhancement. Moreover, the performance of TPCs is compared to other popular codes such as the PCCCs and LDPC codes using equivalent code rates and codeword lengths. The impact of coupling TPCs and hybrid automatic repeat request (HARQ) on the throughput of communications systems is addressed where TPCs error-self detection is used to replace the cyclic redundancy check (CRC) in ARQ-based systems. Finally, the article discusses the main challenges and open research issues that need to be addressed by the research community.

The rest of the paper is organized as follows. TPC construction and decoding are described in Sections II and III, respectively. Using TPCs for joint bit error correction and packet error detection is discussed in Section IV. Section V compares the performance of TPC-based HARQ with conventional HARQ systems. TPCs open research issues are presented in Section VI followed by conclusions in Section VII.

II. Turbo Product Codes Construction

This section discusses TPCs construction methods where the conventional encoding technique is first described. Then, modified construction methods are introduced in Sections II-A–II-F.

Two-dimensional (2D) TPCs are constructed by serially concatenating two linear block codes C^i $(i = 1, 2)$. The two component codes C^i, also referred to as elementary codes, have the parameters $(n_i, k_i, d_{\min}^{(i)})$ which describe the codeword length, number of information bits, and minimum Hamming distance, respectively [46]. To build a product code, $k_1 \times k_2$ information bits are placed in a matrix of k_1 rows and k_2 columns. The k_1 rows are encoded by code C^1 and a matrix of size $k_1 \times n_1$ is generated. Then, the n_1 columns are encoded by the C^2 code and a two-dimensional codeword of size $n_2 \times n_1$ is obtained. The parameters of the product code C are $(n_1 \times n_2, k_1 \times k_2, d_{\min}^{(1)} \times d_{\min}^{(2)})$. The code rate which is the number of information bits divided by the codeword size is calculated as $\zeta = \dfrac{k_1 \times k_2}{n_1 \times n_2}$ for regular TPCs.

Fig. 1 shows an illustration of a TPC codeword. When $n_1 = n_2 \triangleq n$, $k_1 = k_2 \triangleq k$ and $d_{\min}^{(1)} = d_{\min}^{(2)} \triangleq d_{\min}$, a square product code is constructed, denoted as $(n, k, d_{\min})^2$.

TPCs can be constructed using different component codes such as Hamming codes [47]–[49], Bose-Chaudhuri-Hocquenghem (BCH) codes [7], [50], [51], Reed-Solomon (RS) codes [52], and LDPC codes [45], [53]. In addition, several extensions to TPCs classical construction have been proposed in the literature. Examples of these extensions are discussed in Sections II-A–II-F.

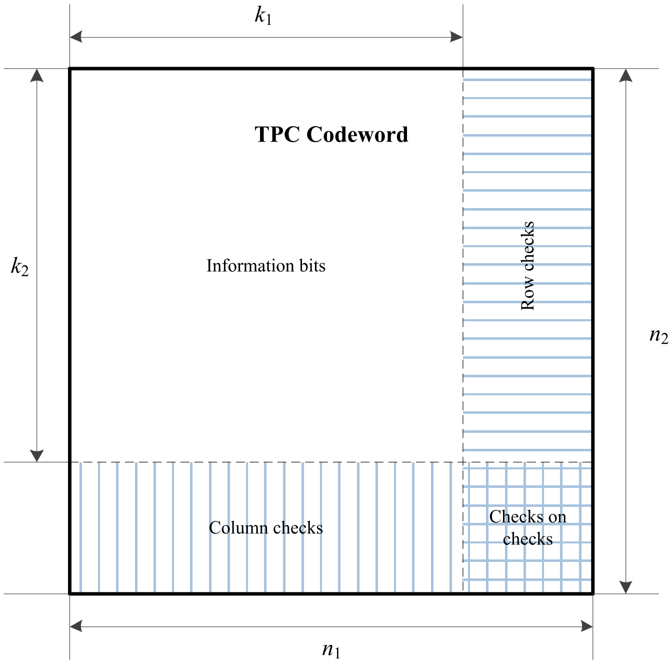

Fig. 1. 2D TPC codeword.

A. Multidimensional TPCs

Multidimensional TPCs can be constructed similar to 2D TPCs, where a particular component code is used for the ith dimension, $i = 1, 2, \ldots, \mathfrak{d}$. In such scenarios, the TPC codeword length, number of information bits and minimum Hamming distance are respectively given by $\mathcal{N} = \prod_{i=1}^{\mathfrak{d}} n_i$, $\kappa = \prod_{i=1}^{\mathfrak{d}} k_i$ and $D_{\min} = \prod_{i=1}^{\mathfrak{d}} d_{\min}^{(i)}$. The code rate of the product code is $\zeta = \frac{\kappa}{\mathcal{N}}$.

Multidimensional TPCs are typically used to construct codes with long codewords and low decoding complexity. The complexity reduction is achieved by using component codes that have short codeword lengths and high code rates. Therefore, performing large number of decoding operations over short codes rather than performing smaller number of decoding operations over long codes. For example, 2D TPCs require $2n$ component code decoding operations per iteration, while 3D TPCs require $3n^2$.

Multidimensional product codes are investigated in [54] and [55] using single parity check (SPC) codes as component codes. The authors in [56] show that the weight (number of non-zero elements) distribution in high code rate TPCs is approximately Gaussian which is a desirable feature to have a good long linear code. They propose construction of low-complexity TPCs with Gaussian distribution by using SPC codes.

In terms of error performance, 2D codes usually outperform higher dimensional codes with the same codeword lengths and code rates [7], [54]. However, the performance difference is not significant. For example, the eBCH$(64, 51, 6)^2$ and eBCH$(8, 7, 2)^4$ TPCs have 4096 codeword length and equivalent code rates. However, the eBCH$(64, 51, 6)^2$ is about 0.4 dB better than the eBCH$(8, 7, 2)^4$ at BER 10^{-5}.

B. Nonbinary TPCs

Non-binary TPCs can be constructed using nonbinary component codes as described in [52] where two RS codes are used as component codes. However, bandwidth efficient coding can be achieved as well using binary codes as component codes followed by multilevel modulation [57]. In both cases, soft decision decoding can be used to minimize the BER.

An example of RS-TPCs is presented in [52] where $k_1 \times k_2$ 2^q-ary information symbols are encoded to obtain $n_1 \times n_2$ TPCs over Galois field GF(2^q). The results given in [52] reveal that RS-TPCs have lower decoding complexity as compared to binary TPCs with equivalent code rates. It is worth noting that the complexity reduction is mostly obtained because RS-TPCs can provide similar error performance for much smaller codeword lengths. For example, RS$(15, 13)^2$ and eBCH$(64, 57)^2$ have equivalent code rates and error performance while the codeword lengths are 900 and 4096 bits, respectively.

C. Modified Row-Column Interleaving

In the literature, additional interleaving processes are introduced on top of the row-column interleaving to reduce the decoding complexity and/or error rate of TPCs. The performance of SPC-TPCs is improved in [58] by passing several SPC-TPCs codewords through an interleaver and a rate-1 recursive convolutional code. However, this introduces additional encoding and decoding delay which may be undesirable for some delay-sensitive applications. The authors in [48] propose rearranging/interleaving the information and parity bits of extended Hamming TPCs in a way which allows identifying the error location directly from the syndrome value. Hence, a syndrome table is no longer needed and decoding complexity is reduced in terms of storage requirements. Error performance is slightly improved over additive white Gaussian noise (AWGN) channels.

In [59], the row-column interleaver is replaced by a constrained uniform interleaver to improve the interleaving gain in two and three dimensional (2D and 3D) TPCs while maintaining the highest minimum Hamming distance. The random interleaver is constrained to have coded bits of every row codeword placed in a different column to maintain the highest minimum Hamming distance of row-column interleaver. However, interleaving gain is improved by independently randomizing bits in each row, then randomizing bits in each column. Improving the interleaving gain, also known as lowering the error coefficient, corresponds to lowering the number of nearest neighbors with minimum Hamming distance [60]. The constrained interleaver size is required to be larger than the row-column interleaver by an integer multiple in order to achieve performance improvement for 2D SPC. A class of interleaved product codes is proposed in [61] where an interleaver is applied only row-wise after row encoding and before column encoding to reserve the high minimum Hamming distance of product codes while reducing the number of low weight codewords. The authors show that this class of TPCs can be viewed as LDPC codes and can be decoded using LDPC decoders.

TABLE I
A SUMMARY OF TPCs MODIFIED CONSTRUCTION METHODS AND THEIR MAIN ADVANTAGES

Method	Related References	Main Advantages
Multidimensional TPCs	[54]–[56]	• Complexity reduction by using component codes with short codeword lengths and high code rates.
Nonbinary TPCs	[52]	• Complexity reduction because nonbinary codes can provide similar error performance to binary codes using smaller codeword lengths.
Modified Row-Column Interleaving	[48], [58], [59], [61]	• Error performance enhancement by improving the interleaving gain. • Complexity reduction by generating codewords which are simpler to decode.
Irregular TPCs	[50], [62], [63]	• Error performance enhancement for some code rates and lengths by using row/column component codes with different code rates.
Extended and Shortened TPCs	[4], [47]	• Error performance enhancement by increasing the minimum Hamming distance. • Enhances flexibility of TPCs in terms of codeword lengths and code rates.
Nonlinear TPCs	[64]	• Error performance enhancement by using nonlinear component codes to provide product codes with larger minimum distance.

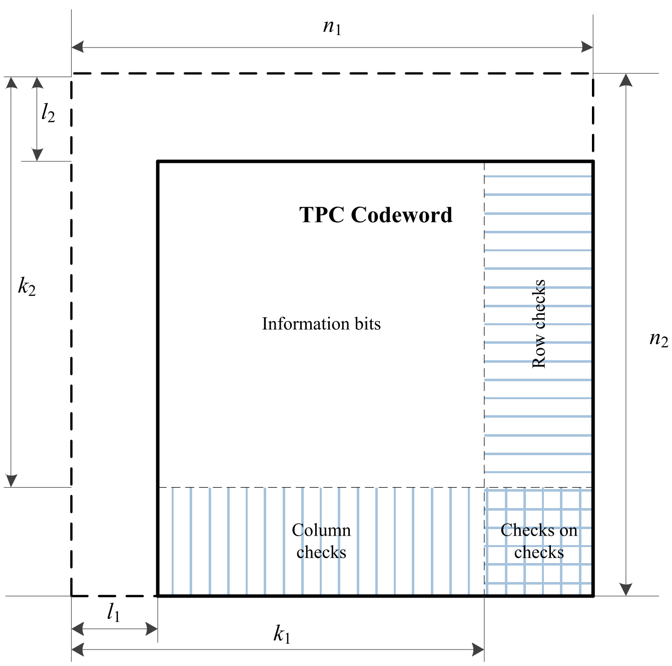

Fig. 2. Shortened TPC codeword.

D. Irregular TPCs

Irregular product codes are constructed in [50], [62], and [63] using row/column component codes with different code rates, but same codeword lengths. These modified product codes are shown to enhance the decoding error probability for some code rates and lengths. Irregular product codes are proposed in [62] where row and/or column codewords have different code rates. Examples are provided for irregular product codes that achieve better error performance than regular TPCs with the same overall code rate over erasure channels. However, this introduces additional encoding and decoding complexity.

E. Extended and Shortened TPCs

An elementary code is extended by adding a single parity bit to the codeword so that the minimum Hamming distance is increased by one. However, the minimum Hamming distance for TPCs increases significantly. For square TPCs, using an extended component code with $d_{\text{min}} = 4$ produces a product code with $d_{\text{min}}^2 = 16$, whereas, an elementary code with $d_{\text{min}} = 3$ produces a product code with $d_{\text{min}}^2 = 9$. Therefore, when using extended component codes to construct TPCs, considerable error rate reduction may be achieved. For some product codes, a coding gain of more than 2 dB is achieved when using extended component codes [4].

On the other hand, shortened TPCs are constructed from component codes C^i with parameters $(n_i - l_i,\ k - l_i,\ d_{\text{min}}^{(i)})$. For $i = 1, 2$, the shortened product codes are generated from encoding $k_1 \times k_2$ information bits using C^1 and C^2; however, the bits in the first l_2 rows and l_1 columns are set to zeros and they are not transmitted. Fig. 2 shows an illustration of a shortened product code. Shortened TPCs are used in IEEE 802.16 [28] to support flexible codeword sizes and code rates. The encoding design of shortened TPCs is discussed in [47].

F. Nonlinear TPCs

The authors in [64] discuss the construction of nonlinear TPCs using two nonlinear component codes or a combination of linear and nonlinear codes. They argue that nonlinear TPCs can have larger minimum distance and hence better error performance. However, iterative decoding used in linear TPCs will not perform well for nonlinear TPCs and hence higher complexity decoders should be invoked. It is worth noting that in nonlinear TPCs the bottom $n_2 - k_2$ rows of the TPCs matrix are not, in general, codewords of the component codes. Table I summarizes TPCs modified construction methods and their main advantages.

III. ITERATIVE DECODING OF TPCS

This section surveys existing literature on TPCs iterative decoding techniques where hard and soft decision decoding methods are described in Sections III-A and III-B, respectively. The performance and complexity trade-off for TPCs decoders is discussed in Section III-C. The main contributions in designing efficient TPCs decoders is summarized in Section III-D.

Finally, the error performance of TPCs is compared to other popular capacity approaching codes in Section III-E.

Turbo product codes are powerful FEC codes that can provide high coding gain. Nevertheless, the complexity of TPCs decoders can be very high when maximum likelihood decoding (MLD) is used. Therefore, sub-optimum iterative decoding methods are alternatively used to reduce the complexity while providing satisfactory performance [7]. Assuming the transmission of binary phase shift keying (BPSK) symbols over an AWGN channel, a transmitted TPC codeword, $\mathbf{U}$, is received as $\mathbf{R} = \mathbf{U} + \mathbf{W}$ where $\mathbf{W}$ is a matrix of AWGN samples with zero mean and $N_0/2$ variance. If hard decision decoding (HDD) is desired, the matrix $\mathbf{R}$ is converted to a binary matrix $\mathbf{B}$ that is fed to the TPC decoder, where $\mathbf{B} = 0.5(\text{sign}[\mathbf{R}] + 1)$ and sign(.) is the signum function. Otherwise, $\mathbf{R}$ is fed directly to the decoder for soft decision decoding (SDD).

A. Hard-Input Hard-Output Decoding

In hard-input hard-output (HIHO) decoding, the code matrix $\mathbf{B}$ is partitioned into row/column vectors which are decoded independently using conventional HDD techniques such as the Berlekamp-Massey algorithm [65]–[67]. Assuming that we start decoding the rows of $\mathbf{B}$, which corresponds to the first half iteration, the second half iteration is performed on the columns of the matrix obtained from the previous half iteration. This process is repeated several times so that residual errors at one iteration may be corrected in the succeeding ones. The error patterns that are not corrected by the first few iterations are permanent errors known as closed-chain errors [68], [69]. The decoding process is terminated when the maximum number of iterations ($\mathcal{I}_{\text{max}}$) is reached, or if all rows and columns are valid codewords of their respective elementary codes.

1) Non-Sequential Decoding: In conventional HIHO decoding, the row/column decoding is performed sequentially. However, in [70] a non-sequential decoding algorithm is proposed to improve the performance of HIHO decoding. The algorithm utilizes the reliability information embedded in the received code components to avoid error amplification. Based on the reliability of each component codeword in the received TPC matrix, a decision is made on whether to decode or to leave that code component without decoding. Simulation results show that the proposed algorithm offers a substantial improvement over traditional sequential decoding with negligible additional complexity.

2) Closed-Chains Error Correction: Fig. 3 shows an example of a closed-chain error pattern in $(8, 4, 4)^2$ TPC. The shaded cells with $\times$ marks correspond to 4 bit errors in $\mathbf{B}$. For this example, the minimum Hamming distance of the product code is $d_{\text{min}}^2 = 16$ and the error correction capability of MLD is $t = \lfloor \frac{d_{\text{min}}^2 - 1}{2} \rfloor = 7 > 4$ where $\lfloor . \rfloor$ is the floor function. Therefore, $\mathbf{B}$ will be decoded successfully using MLD. However, the iterative decoder will not be able to correct the error pattern using component codes with $t = \lfloor \frac{d_{\text{min}} - 1}{2} \rfloor = 1 < 2$.

A HIHO decoding algorithm is proposed in [46] to correct closed-chains error patterns in HIHO turbo product codes. The proposed technique is based on correlating the horizontal and vertical component codes to estimate the location of the erroneous bits in the closed-chain of errors. Erasure decoding is then used to correct the identified bit errors. For particular codes, a noticeable coding gain improvement of about 1.5 dB can be achieved when compared to the standard sequential HIHO decoding and about 0.8 dB when compared to the non-sequential HIHO decoding.

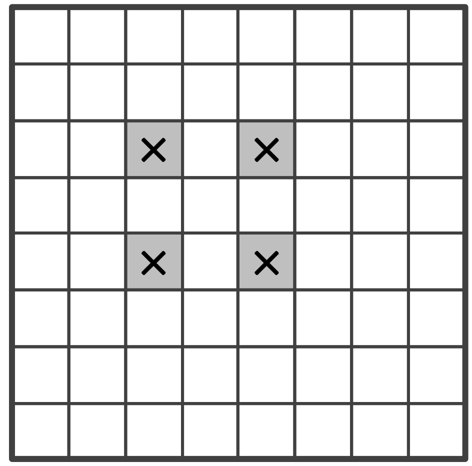

Fig. 3. Closed-chain error-pattern example for $(8, 4, 4)^2$ TPC.

3) HIHO With Symbol Reliability: In [71], the HIHO decoder employs a reliability metric for each element in the decoded TPC matrix. Assuming BPSK, the decision of each decoded element may be flipped if its reliability remained below a threshold for a number of iterations. This reliability-based approach is proposed to improve the performance of HIHO decoders for multidimensional single parity TPCs. The threshold value may change over successive iterations. Some threshold values were proposed based on simulations for the considered codes. The algorithm requires additional memory compared to conventional HIHO decoding.

B. Soft-Input Soft-Output Decoding

Near-optimum decoding of TPCs is achieved by performing a number of soft-input soft-output (SISO) iterative decoding processes. Similar to HIHO, the TPC matrix is partitioned into smaller row/column vectors which are individually decoded using a soft decision iterative decoding algorithm.

1) Pyndiah-Chase-II Algorithm: Pyndiah-Chase-II algorithm is based on SISO iterative decoding where the rows and columns of $\mathbf{R}$ are decoded individually using SDD. During the first half iteration, actual soft information is available from the demodulator; however, extrinsic information is used in the succeeding iterations. The SDD is implemented using the Chase-II decoder [72] which performs a limited search for the maximum likelihood component codeword instead of a prohibitively complex exhaustive search.

The search process can be described as follows.

a) The least reliable p bits in $\mathbf{b} = (b_1, b_2, \ldots, b_n)$ (a component codeword in $\mathbf{B}$) are marked using $\mathbf{r} = (r_1, r_2, \ldots, r_n)$ (a component codeword in $\mathbf{R}$). In stationary AWGN channels, the normalized reliability is given by $|r_i|$.

b) 2^p different error patterns are generated using the marked p bits in $\mathbf{b}$. An error pattern is a vector whose entries are all zeros except the entries marked in the previous step. 2^p different error patterns are generated by altering the values of the marked p bits.

c) 2^p different test patterns are generated by adding each error pattern to $\mathbf{b}$. Each of the 2^p test patterns is decoded using HDD to produce 2^p candidate codewords. The successful candidate codeword $\mathbf{d}$ is the one that has the minimum Euclidean distance to $\mathbf{r}$. Therefore, the number of HDDs performed in each iteration is $2^p(n_1 + n_2)$.

Once the first half iteration is completed, the Chase-II decoder output is the binary vector $\mathbf{d}$; hence, we still need to generate soft information for each bit in $\mathbf{d}$ to enable SDD for the next iterations. Note that the elements of $\mathbf{d}$ are mapped from $\{0, 1\}$ to $\{-1, +1\}$. The soft information after the first iteration is calculated using

$$\tilde{\mathbf{r}}(m) = \mathbf{r} + \alpha(m)\mathbf{w}(m) \tag{1}$$

where $\tilde{\mathbf{r}}(m)$ is the soft data fed to the Chase-II decoder at the mth iteration, $\mathbf{r}$ is the demodulator soft output, $\alpha(m)$ is a scaling factor obtained experimentally, and $\mathbf{w}(m)$ is the extrinsic information calculated from the previous iteration. Extrinsic information is computed as

$$\mathbf{w}(m+1) = \grave{\mathbf{r}}(m) - \tilde{\mathbf{r}}(m), \tag{2}$$

where

$$\grave{\mathbf{r}}(m) = \frac{1}{4}\left(\left|\tilde{\mathbf{r}} - \mathbf{d}^{(2)}\right|^2 - \left|\tilde{\mathbf{r}} - \mathbf{d}^{(1)}\right|^2\right) \times \mathbf{d}^{(1)} \tag{3}$$

and $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$ are the closest and next closest candidate codewords to $\tilde{\mathbf{r}}$, respectively. For each bit i in $\grave{\mathbf{r}}(m)$, $\mathbf{d}^{(2)}$ is chosen such that $\mathbf{d}^{(2)} \neq \mathbf{d}^{(1)}$ at the ith bit, $i \in \{1, 2, \ldots, n\}$. In cases where it is not possible to find $\mathbf{d}^{(2)}$ we use

$$\mathbf{w}(m+1) = \beta(m) \times \mathbf{d}, \quad \beta \geq 0 \tag{4}$$

where β is also a scaling factor and $\mathbf{d} = \mathbf{d}^{(1)}$. The values of α and β for 8 half iterations ($m \in \{1, 2, \ldots, 8\}$) are given in [7],

$$\alpha = [0.0, 0.2, 0.3, 0.5, 0.7, 0.9, 1.0, 1.0] \tag{5}$$

and

$$\beta = [0.2, 0.4, 0.6, 0.8, 1.0, 1.0, 1.0, 1.0]. \tag{6}$$

Entries in the extrinsic information matrix computed using (2) are normalized to have a mean absolute value of one [7]. Fig. 4 shows a general block diagram for Pyndiah-Chase-II decoder. Moreover, a structured summary of the decoding algorithm is shown in Table II. A MATLAB implementation of the TPC encoder and Pyndiah-Chase-II decoder are available online in [73].

2) Other SISO Algorithms: A simplified adaptive belief propagation algorithm for TPCs SISO decoding is proposed in [74] as an alternative to Pyndiah-Chase-II Algorithm. The proposed decoder provides similar performance with a claimed high degree of parallelism. In [75], Kaneko's [76] algorithm is used for soft-input hard-output decoding followed by reliability calculations to convert the hard output to a soft output.

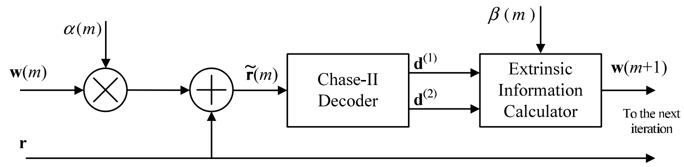

Fig. 4. General block diagram for Pyndiah-Chase-II decoder.

TABLE II
PYNDIAH-CHASE-II DECODING ALGORITHM FOR SQUARE TPCS ASSUMING BPSK MODULATION

Input: $\mathbf{R}, \mathcal{I}_{\max}, \alpha, \beta$
Initialize: $\mathbf{w} = 0$
for $m = 1$ to $\mathcal{I}_{\max}$ **do**
  for $v = 1$ to 2 **do**
    for $i = 1$ to n **do**
      $\mathbf{r} = \mathbf{R}(i, :)$
      $\tilde{\mathbf{r}}(m) = \mathbf{r} + \alpha(m)\mathbf{w}(m)$
      $\mathbf{b} = 0.5\,(\mathrm{sign}\,[\tilde{\mathbf{r}}(m)] + 1)$
      Mark the least reliable p bits in $\mathbf{b}$ using $|\tilde{r}_j|$, $j \in \{1, 2, ..., n\}$
      Generate 2^p error patterns using the marked p bits
      Generate 2^p test patterns by adding each error pattern to $\mathbf{b}$.
      Decode each test pattern using HDD to produce 2^p candidates
      Compute Euclidean distance of each candidate codeword to $\tilde{\mathbf{r}}$
      Find $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$, set $\mathbf{d} = \mathbf{d}^{(1)}$
      if not possible to find $\mathbf{d}^{(2)}$ **then**
        Compute $\mathbf{w}(m+1)$ using (4)
      else
        Compute $\mathbf{w}(m+1)$ using (1), (2) and (3)
      end if
      $\tilde{\mathbf{R}}(i, :, m) = \tilde{\mathbf{r}}(m)$
      $\mathbf{D}(i, :) = \mathbf{d}$
    end for
    $\mathbf{R} = \mathbf{R}^T$ $\rightarrow (.)^T$ is the transpose function
    $\tilde{\mathbf{R}}(m) = \tilde{\mathbf{R}}(m)^T$
    $\mathbf{D}^T$
  end for
end for
return $\mathbf{D}$

Kaneko's algorithm tests error patterns sequentially until a decoding condition is met and remaining error patterns are not tested. Due to this adaptive nature of Kaneko's algorithm a complexity reduction in SISO decoding is possible. Order-i reprocessing algorithm [77] is used in [78] for SISO decoding. Additional decoding steps are proposed to correct common residual errors which could not be corrected by the iterative decoder. This improves the performance at low bit error rate, however, at the cost of increased complexity.

The authors in [79] propose a SISO decoder based on an alternative method for component decoding known as Bidirectional Efficient Algorithm for Searching Code Trees (BEAST). The authors show that their proposed SISO decoder can achieve better performance than Pyndiah-Chase-II decoder. However, did not provide a complexity comparison of the two schemes. They pointed out that direct complexity comparison is not feasible on a general level. An algorithm is proposed in [80] to reduce complexity of trellis-based optimum decoding for turbo product codes constructed by serial concatenation of a single parity check code and a low dimensionality binary linear block code with small k_2. Nevertheless, the obtained results reveal that optimum decoding achieves a small coding gain (0.1 dB) over sub-optimum iterative decoding for the considered class of TPCs. An additional coding gain of 0.4 dB can be achieved when using a modified construction of serially concatenated codes which are not considered to be TPCs.

In [81], a decoding algorithm is proposed for analog product codes using linear programming. However, the system performance is evaluated using loose upper bounds and the provided simulation results are based on the assumption that erroneous decoding happens when there are t rows with t errors where t is the error correction capability of the component code. TPCs decoders may be able to correct such errors unless they are arranged in closed chains.

The SISO decoding algorithms has been adapted for different application-specific channel models. In [82]–[84], the turbo decoder is adapted and its performance is evaluated for channels with partial-band interference, whereas, partial-time jamming is considered in [85]. The performance of TPCs is also optimized for magnetic recording partial response channels and non-Gaussian optical channels in [86] and [31], respectively.

C. Complexity and Performance Trade-Off

SISO decoders require considerable computational power as compared to HIHO decoders. However, in the literature, the high computational complexity of SISO decoding is justified by its high coding gain advantage over HIHO decoding as shown by the BER performance curves in Fig. 5 and Fig. 6. The BER performance of some extended BCH (eBCH) product codes over AWGN channels is shown in Fig. 5, whereas, the performance over independently and identically distributed (i.i.d) Rayleigh fading is shown in Fig. 6. In the figures, Pyndiah-Chase-II algorithm with $p = 4$ is used for SISO decoding, and the maximum number of iterations $\mathcal{I}_{\max} = 4$ for both SISO and HIHO.

Moreover, bandwidth efficient coding can be achieved using TPC followed by multilevel modulation. The soft decision decoding is slightly modified when multilevel modulation schemes are used [57]. Fig. 7 shows the BER performance of $\mathcal{M}$-ary quadrature amplitude modulation (QAM) systems coded using eBCH$(128, 120, 4)^2$ TPC. The results are shown for the SISO decoder described in [57].

The complexity of SISO and HIHO TPCs decoders is mainly determined by the number of HDD operations performed [46]. For SISO Pyndiah-Chase-II decoder, the number of HDD operations per half iteration used to decode the received matrix $\mathbf{R}$ is $n \times 2^p$. For HIHO decoding, the number of HDD operations per half iteration is n. The complexity of each HDD operation depends on the adopted HDD algorithm. For example, complexity analysis of BCH-HDD is given in [87]. The computational complexity for each HDD operation N_{HDD} is bounded by

$$45\lambda^2 n^2 \left(\log_{10} n\right)^2 < N_{\text{HDD}} < (45\lambda + 4)\lambda n^2 \left(\log_{10} n\right)^2 \quad (7)$$

where $\lambda = t/n$, and t is the number of errors that can be corrected by the code.

TPCs decoders provide a trade-off between performance and complexity. In particular, the probability of successful decoding increases by increasing the maximum number of iterations

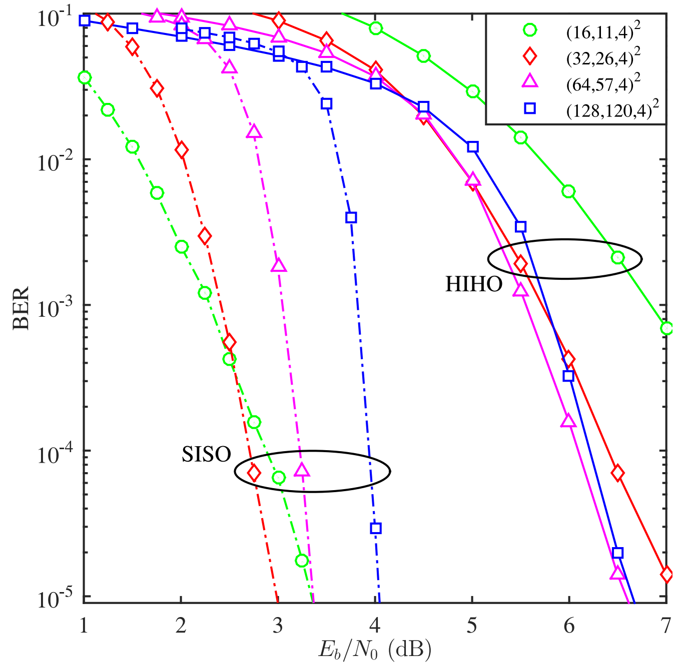

Fig. 5. BER of eBCH-TPC using HIHO and SISO decoding with $p = 4$ and $\mathcal{I}_{\max} = 4$ over AWGN channels.

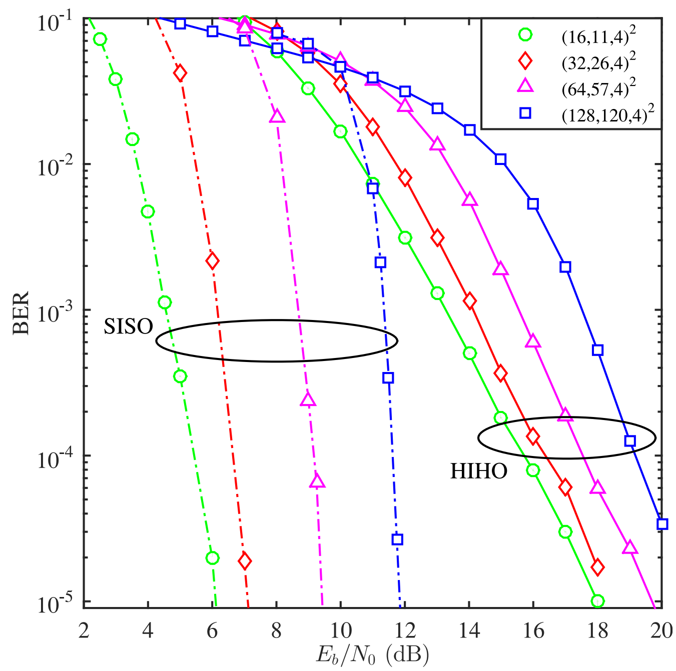

Fig. 6. BER of eBCH-TPC using HIHO and SISO decoding with $p = 4$ and $\mathcal{I}_{\max} = 4$ over Rayleigh fading channels.

$\mathcal{I}_{\max}$, and the size of search space which is controlled by p. However, the complexity of SISO decoders increases exponentially as a function of p. In [4] and [7] it is shown that no significant performance gain is achieved for $\{p, \mathcal{I}_{\max}\} > 4$.

D. Efficient TPCs Decoding

Several algorithms have been proposed to improve the efficiency of TPCs decoders. A fast implementation of Chase-II

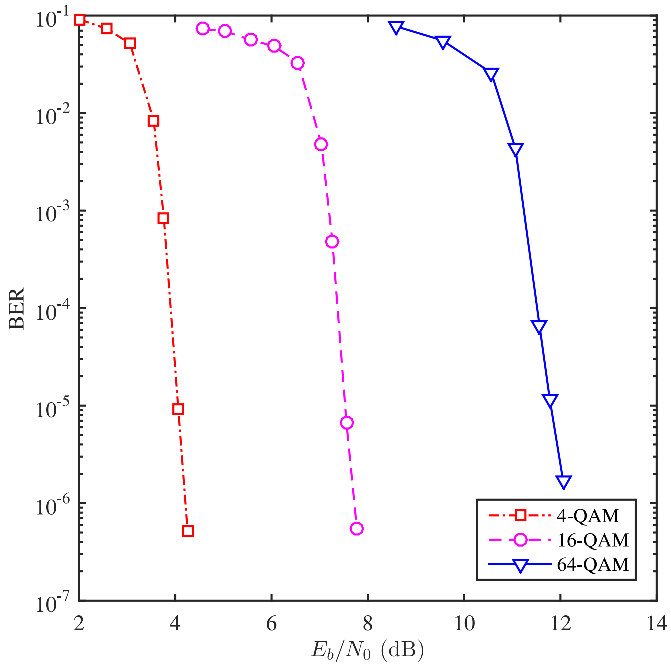

Fig. 7. BER of $\mathcal{M}$-ary QAM systems with eBCH(128, 120, 4)^2 over AWGN channels. SISO decoding is used with $p = 4$ and $\mathcal{I}_{\max} = 4$.

algorithm is proposed in [88] by processing test patterns in a particular order to have less number of patterns being tested. Moreover, the proposed re-ordering enables the derivation of a sequence syndrome from other sequences recursively. The efficiency of such decoder is demonstrated for single error correction codes. The complexity of Pyndiah-Chase-II decoder is also reduced in [89] for single error correcting codes by obtaining the syndromes, even parities and the extrinsic information using fewer operations. The authors in [90] proposed reducing the number of decoded test patterns in the Chase-II algorithm to reduce decoding complexity. Syndromes are exploited to classify the test patterns so that test patterns with the same codeword are not unnecessarily decoded.

An efficient decoder for TPCs is proposed in [91] where the new decoder is composed of a standard SISO decoder with a small number of iterations followed by a HIHO decoder. The proposed decoder is implemented by configuring the conventional SISO decoder to operate in two modes, a SISO mode and a HIHO mode. The performance of the hybrid decoder is investigated using different TPCs with different parameters over AWGN channels. Compared to a standard SISO, the hybrid decoder reduces the overall computational complexity while providing comparable BER performance. The hybrid decoder also provides a substantial flexibility to optimize the performance/complexity trade-off.

The work in [92] proposes a method to compute the extrinsic information from one HDD rather than 2^p HDD. A row/column input codeword is hard decision decoded and the number of possible errors e in the input row/column codeword is computed. The extrinsic information is then estimated based on e and $d_{\min}$. This method is designed for reliable HDD, i.e., high SNR. Therefore, the proposed decoder resorts to Pyndiah-Chase-II algorithm when e is greater than a specified threshold. Decoding complexity is reduced at moderate and high SNRs while error performance remains almost unchanged.

In [93] a hybrid decoder is proposed to improve error performance with low computational burden for product codes with large ratio of minimum Hamming distance to code rate. The authors propose concatenating a soft-input hard-output (SIHO) decoder to Pyndiah-Chase-II decoder at the last iteration. SIHO is a two step decoder where a hard-limited reliability metric is passed from row decoding to column decoding.

The major contributions that aim at enhancing TPCs decoders in terms of complexity and error performance are summarized in Table III.

E. Performance Comparison With Other Capacity-Approaching Codes

In addition to TPCs, there are other popular capacity approaching codes such as PCCCs and LDPC codes. Fig. 8 compares the BER performance of these codes over AWGN channels. The comparison is performed using equivalent code rates and codeword lengths. The number of decoding iterations for the three coding schemes is chosen so that no significant performance gain is achieved if additional iterations are performed. In the figure legend, the codes are labeled and the corresponding parameters $\{\mathcal{N}, \zeta, \mathcal{I}_{\max}\}$ for each code are shown in Table IV.

The TPCs are constructed using eBCH$(n, k, d_{\min})^2$ to produce codes with $\{\mathcal{N} = n^2, \zeta = \frac{k^2}{n^2}, \mathcal{I}_{\max} = 4\}$. On the other hand, PCCCs are implemented using two identical recursive systematic convolutional encoders. The constituent encoders used in this example have parameters $l_{\mathrm{i}} = 1$, $l_{\mathrm{o}} = 2$, $l_{\mathrm{k}} = 4$ where l_{i}, l_{o} and l_{k} denote the number of input bits, number of output bits and the constraint length, respectively. The generator polynomials and the feedback connection polynomial in octal form are $G_1 = 13$, $G_2 = 15$ and $G_{\mathrm{f}} = 13$, respectively. The first encoder operates on the input information bits directly, while the second encoder operates on interleaved information bits. The PCCC encoder basic code rate is $\zeta = 1/3$. The length of information bits and the puncturing periods are selected to generate PCCCs with codeword lengths and code rates equivalent to their counterpart TPCs. The PCCCs decoder performs a maximum of $\mathcal{I}_{\max} = 4$ soft decision decoding iterations using the max-log approximation algorithm [94].

LDPC codes are also constructed with equivalent parameters using the methods described in [95] and [96]. The generated LDPC codes are soft decision decoded with $\mathcal{I}_{\max} = 50$, beyond which no significant performance gain is achievable.

The three codes provide similar error performance for low and moderate code rates. However, for high code rates PCCCs do not perform well due to high amount of puncturing, e.g., PCCC 4 {16384, 0.87, 4} in the right subfigure of Fig. 8. Although there are some techniques which can be used to improve the performance of PCCCs at high code rates, the improvement is not significant and the decoder complexity is increased [97], [98].

TABLE III
MAJOR CONTRIBUTIONS IN ENHANCING TPCS DECODERS IN TERMS OF COMPLEXITY AND ERROR PERFORMANCE

Category	Year	Author(s)	Contribution
Complexity Reduction	1998	Pyndiah [7]	Introduced a SISO decoder based on Chase-II algorithm.
	2001	Hirst *et al.* [88]	Proposed a fast implementation of Chase-II algorithm which reduces the number of tested patterns and uses efficient recursive syndrome computation.
	2001	Dave *et al.* [75]	Kaneko's algorithm is used in SISO decoding.
	2004	Argon and McLaughlin [89]	Obtained the syndromes, even parities and the extrinsic information of Pyndiah-Chase-II algorithm using fewer operations.
	2007	Xu *et al.* [47]	The extrinsic information computation is modified for lower decoding complexity of shortened TPCs.
	2008	Xu *et al.* [48]	Storage requirement is reduced where a syndrome table is no longer needed.
	2009	Chen *et al.* [90]	Syndromes are exploited to classify the test patterns so that test patterns with the same codeword are not unnecessarily decoded.
	2009	Al-Dweik *et al.* [91]	An efficient hybrid decoder which operates in SISO and HIHO modes is proposed.
	2014	Lu *et al.* [92]	A method is proposed to compute the extrinsic information from one HDD rather than 2^p HDD.
Error Performance Enhancement	1984	Hirasawa *et al.* [50]	Irregular TPCs are constructed using component codes with different code rates, but same codeword lengths to enhance the performance for some codes.
	2004	Li *et al.* [58]	The performance of SPC-TPCs is improved by passing several SPC-TPCs codewords through an interleaver and a rate-1 recursive convolutional code.
	2004	Martin *et al.* [78]	Order-i reprocessing algorithm is used for SISO decoding to improve the performance at high SNR.
	2008	Loncar *et al.* [79]	Proposed bidirectional efficient algorithm for searching code trees (BEAST).
	2009	Al-Dweik and Sharif [70]	A non-sequential decoding algorithm is proposed to improve the performance of HIHO decoding.
	2011	Al-Dweik and Sharif [46]	A HIHO decoding algorithm is proposed to correct closed-chains error patterns in TPCs.
	2012	Baldi *et al.* [61] and Fonseka *et al.* [59]	The row-column interleaver is replaced by constrained random interleavers.
	2014	Ravid and Amrani [93]	Proposed concatenating a SIHO decoder to Pyndiah-Chase-II decoder at the last iteration.
	2015	Janvars and Farkas [71]	A reliability-based approach is proposed to improve the performance of HIHO decoders for multidimensional SPC-TPCs.

TABLE IV
PARAMETERS $\{\mathcal{N}, \zeta, \mathcal{I}_{\max}\}$ FOR THE CODES USED IN FIG. 8

	$\{\mathcal{N}, \zeta, \mathcal{I}_{\max}\}$		
	TPC	PCCC	LDPC
1	{256,0.47,4}	{256,0.48,4}	{256,0.47,50}
2	{1024,0.66,4}	{1036,0.66,4}	{1024,0.66,50}
3	{4096,0.79,4}	{4108,0.80,4}	{4096,0.79,50}
4	{16384,0.89,4}	{16384,0.87,4}	{16384,0.89,50}

LDPC codes error performance is also affected by the codeword length. For relatively long codewords, LDPC codes have a slight advantage over other codes. However, for short codeword lengths this advantage vanishes and LDPC codes may provide error performance worse than other codes. For example, TPC 1 $\{256, 0.47, 4\}$ and PCCC 1 $\{256, 0.48, 4\}$ have a performance advantage of ~ 0.6 dB and 0.4 dB, respectively, over LDPC 1 code $\{256, 0.47, 50\}$ at BER of 10^{-5} as shown in the left subfigure of Fig. 8.

IV. TPCS ERROR SELF-DETECTION

This section discusses TPCs error self-detection where two main methods are presented in Sections IV-A and IV-B. The complexity of TPCs self-detection is compared to conventional error detection techniques in Section IV-C. TPCs self-detection performance is evaluated in terms of false alarm and misdetection in Section IV-D where numerical examples are provided.

Error detection is an essential process in TPCs because it can be used for early stopping of the iterative decoding process to reduce the complexity and delay. Furthermore, it can be used to report that the decoding process was unsuccessful, and hence, it can be used in automatic repeat request (ARQ) systems, or in similar applications. Error detection is usually performed using CRC codes which can be realized using different software and hardware implementations [99]–[102]. However, the CRC process can have adverse effects on the computational complexity, delay, throughput and energy consumption for systems with continuous and high data rate transmission [22], [103]–[107].

A. Syndrome Error Checking (SEC) Self-Detection

The TPCs codeword has a matrix structure which lends itself to error self-detection with relatively low complexity and no additional CRC redundancy [4]. In the iterative decoding of TPCs, all rows of the code matrix are decoded sequentially

TABLE V
THE RELATIVE COMPLEXITY OF TPC ERROR DETECTION TECHNIQUES TO 16-BIT CRC-8005

Relative Complexity	$(128, 120, 4)^2$		$(64, 57, 4)^2$		$(32, 26, 4)^2$		$(16, 11, 4)^2$	
	LB	UB	LB	UB	LB	UB	LB	UB
SEC	0.0030	0.3575	0.0063	0.3589	0.0141	0.3658	0.0374	0.4116
PEC	0.0013	0.1514	0.0028	0.1587	0.0067	0.1745	0.0204	0.2245

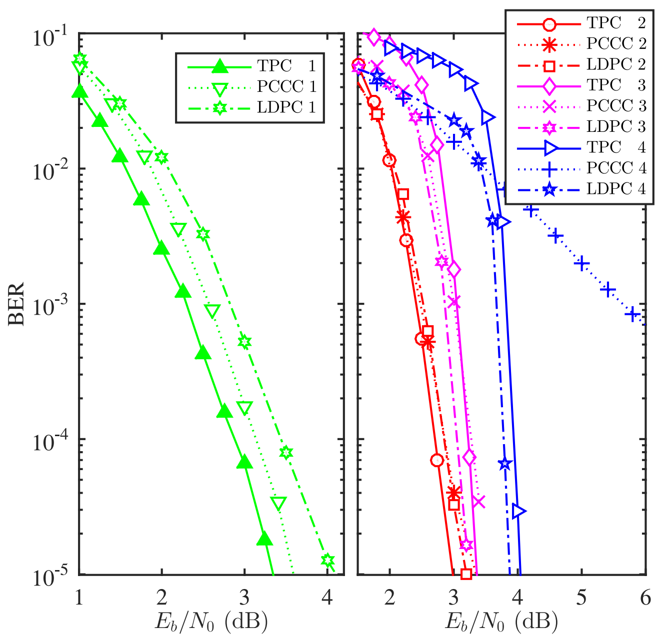

Fig. 8. BER comparison of TPCs, PCCCs and LDPC codes over AWGN channels using equivalent code parameters. Table IV shows the parameters $\{\mathcal{N}, \zeta, \mathcal{I}_{\max}\}$ for each labeled code in the legend.

followed by column decoding. If the error correction process is successful, all rows and columns will be valid codewords of their respective elementary code C_i. Consequently, error detection after the last column decoding process can be performed using syndrome error checking (SEC) of row codewords. If all syndromes are equal to zero, then the TPC codeword is declared error free [4], [22].

In general, the last column decoding process guarantees that all columns in the matrix are valid codewords. However, if the row/column decoding process does not necessarily produce a valid codeword [46], then the last two half iterations should be checked.

B. Parity Error Checking (PEC) Self-Detection

For TPCs with extended component codes, self-detection can be performed by exploiting the parity check bits used for extending the component codes [108]. Assuming even parity is used to construct the extended component codes, the received TPC matrix is declared erroneous once an odd parity row is found. On the other hand, if all information rows have even parity, the TPC packet is declared error-free. In the following Sections IV-C and IV-D, the relative complexity and error detection performance of the SEC and PEC methods will be described.

C. Self-Detection Relative Complexity

Complexity analysis in [22] and [108] demonstrates the significant complexity reduction can be achieved by TPCs self-detection as compared to CRC-based systems.

TPCs self-detection and CRC decoding can be implemented using only exclusive-OR (XOR) operations. Therefore, comparing the computational complexity of the TPCs self-detection to other CRC-based detection standards can be achieved using the relative complexity, which is the ratio of the number of operations in TPCs self-detection to the number of operations in CRC decoding.

For TPCs error self-detection, the number of operations depends on the used component code. Moreover, it varies randomly based on the index of first row that has a detectable error. Because the row syndromes are computed sequentially, once a non-zero row syndrome is found the self-detection process is aborted and the received matrix is declared erroneous. If no error is detected in all information rows, the self-detection process is aborted and the TPC packet is declared error-free. Therefore, the complexity of TPCs self-detection is expressed using upper and lower bounds, UB and LB, respectively.

Table V shows the relative complexity UB and LB of TPCs self-detection when SEC and PEC are used for different eBCH TPCs, namely, $(128, 120, 4)^2$, $(64, 57, 4)^2$, $(32, 26, 4)^2$ and $(16, 11, 4)^2$. The complexity is computed relative to 16-bit CRC-8005 which is a commonly used CRC code. It can be observed that both TPCs self-detection methods are remarkably less complex than CRC detection. The relative complexity of self-detection is reduced by almost 50% when using PEC instead of SEC.

It should be noted that the complexity reduction is proportional to the number of CRC bits and the TPC codeword size. For example, the relative complexity of SEC-based self-detection for eBCH$(128, 120, 4)^2$ is bounded by 6×10^{-4} and $\mathcal{C}_S \leq 0.0715$ of the 32-bit CRC-1EDC6F41 [22].

D. False Alarm and Misdetection Rates

When performing error detection at the receiver side, an error can be misdetected if a row in the decoded matrix is a valid codeword of C^i but not equal to its corresponding transmitted row codeword. Moreover, a correct TPC codeword can be declared as erroneous, generating a false alarm, if all the information bits are correct but the row parity check bits contain errors. The performance of the CRC error detection and TPCs self-detection schemes is evaluated in terms of

TABLE VI
FALSE ALARM RATE OF TPCS SELF-DETECTION VERSUS 16-BIT CRC DETECTION IN RAYLEIGH FADING CHANNELS WHEN SISO DECODING IS USED. THE HYPHEN '–' MEANS THAT THE NUMBER OF ERROR-FREE PACKETS AFTER TRANSMITTING 10^6 PACKETS IS NOT ENOUGH TO COMPUTE THE FAR RELIABLY. THE SYMBOL $\langle 10^{-5} \rangle$ MEANS FAR $< 10^{-5}$

E_b/N_0	$(16, 11, 4)^2$		$(32, 26, 4)^2$	
(dB)	TPC	CRC	TPC	CRC
0	3.3×10^{-1}	–	–	–
1	2.9×10^{-1}	1.3×10^{-1}	–	–
2	1.4×10^{-1}	8.2×10^{-2}	–	–
3	7.5×10^{-2}	4.2×10^{-2}	–	–
4	2.8×10^{-2}	1.8×10^{-2}	–	–
5	7.4×10^{-3}	6.5×10^{-3}	6.1×10^{-2}	2.3×10^{-3}
6	9.9×10^{-4}	1.4×10^{-3}	1.8×10^{-2}	8.4×10^{-4}
7	9.0×10^{-5}	1.7×10^{-4}	2.9×10^{-3}	1.3×10^{-4}
8	$\langle 10^{-5} \rangle$	$\langle 10^{-5} \rangle$	1.4×10^{-4}	$\langle 10^{-5} \rangle$
9	$\langle 10^{-5} \rangle$	$\langle 10^{-5} \rangle$	$\langle 10^{-5} \rangle$	$\langle 10^{-5} \rangle$

TABLE VII
MISDETECTION RATE OF TPCS SELF-DETECTION VERSUS 16-BIT CRC DETECTION IN RAYLEIGH FADING CHANNELS WHEN SISO DECODING IS USED. THE HYPHEN '–' MEANS THAT THE NUMBER OF ERRONEOUS PACKETS AFTER TRANSMITTING 10^6 PACKETS IS NOT ENOUGH TO COMPUTE THE MDR RELIABLY. THE SYMBOL $\langle 10^{-5} \rangle$ MEANS MDR $< 10^{-5}$

E_b/N_0	$(16, 11, 4)^2$		$(32, 26, 4)^2$	
(dB)	TPC	CRC	TPC	CRC
0	1.2×10^{-4}	2.3×10^{-5}	$\langle 10^{-5} \rangle$	1.3×10^{-5}
1	5.6×10^{-4}	1.4×10^{-5}	$\langle 10^{-5} \rangle$	1.3×10^{-5}
2	4.3×10^{-3}	1.1×10^{-5}	$\langle 10^{-5} \rangle$	1.7×10^{-5}
3	1.7×10^{-2}	2.8×10^{-5}	$\langle 10^{-5} \rangle$	1.9×10^{-5}
4	4.7×10^{-2}	2.4×10^{-5}	$\langle 10^{-5} \rangle$	1.9×10^{-5}
5	1.0×10^{-1}	2.7×10^{-5}	2.0×10^{-4}	2.2×10^{-5}
6	1.8×10^{-1}	1.8×10^{-5}	3.3×10^{-3}	1.3×10^{-5}
7	3.7×10^{-1}	–	1.9×10^{-2}	1.7×10^{-5}
8	4.6×10^{-1}	–	8.4×10^{-2}	–
9	–	–	3.6×10^{-1}	–

false alarm rate (FAR) and misdetection rate (MDR) in [22]. The FAR is the number of false alarms divided by the number of correct packets, whereas, the MDR is the number of misdetections divided by the number of erroneous packets. Table VI and VII show the FAR and MDR, respectively, for eBCH TPCs for different E_b/N_0 values in Rayleigh fading channels. E_b is the average energy per bit and N_0 is the noise power spectral density.

Table VI shows that the FAR rate decreases as the channel SNR increases. A false alarm happens when the information bits are correct but the parity bits related to error detection are in error. Therefore, when the channel SNR increases the probability of having an error in the parity bits but not in the information bits reduces and the FAR subsequently decreases. The table also shows that 16-bit CRC error detection has lower FAR than $(32, 26, 4)^2$ TPC self-detection because the number of parity bits of the TPCs is larger and hence the probability of error within these bits is larger as well.

Typically, a high number of false alarms decreases the system throughput since correct packets may be rejected. However, for TPCs self-detection when the FAR is relatively high the probability of having correct packets is low. Hence, FAR of TPCs self-detection has negligible effect on the system throughput. On the contrary, TPCs self-detection eliminates the additional CRC bits and subsequently improves the system performance not only in terms of throughput, but also in terms of complexity, delay and energy consumption.

Table VII shows that for $(32, 26, 4)^2$, the MDR of TPC self-detection is smaller than CRC detection at low SNR values. The MDR of TPCs self-detection deteriorates as the TPC codeword size is reduced. Small TPCs codewords have less number of component codes involved in TPCs self-detection as compared to larger TPCs codewords. Therefore, the joint error detection capability of these component codes decreases when smaller TPCs codewords are used. The MDR of TPCs self-detection is smaller than 16-bit CRC code for all SNR values when large TPCs are used such as $(128, 120, 4)^2$ [22].

Moreover, for small TPC codeword sizes, we observe that the MDR of TPCs self-detection increases as the SNR increases. At low SNR values, the number of errors in the decoded TPCs is high with various patterns; therefore, the probability of having undetected errors in all component codewords is small. As the channel SNR increases, the number of errors decreases; however, the errors tend to have closed chain patterns. These types of errors are likely to be misdetected in all affected component codewords if the error pattern is larger than their individual error detection capability. Hence, the TPCs misdetection probability increases. However, it should be noted that at high SNR the probability of having an erroneous packet is low which alleviates the effect of high MDR on the system throughput. On the other hand, the MDR of CRC detection is almost constant and matches the theoretical approximation $2^{-16} \approx 1.53 \times 10^{-5}$.

The performance of TPCs self-detection in terms of MDR and FAR using PEC is studied in [108]. Fig. 9 shows that PEC and SEC methods provide similar FAR performances. Moreover, Fig. 10 shows that SEC has a better MDR performance than PEC-based self-detection. However, the PEC method has lower computational complexity than SEC as illustrated in Section IV-C.

V. TPCS-BASED HYBRID AUTOMATIC REPEAT REQUEST (TPCS-HARQ)

This section discusses adopting TPCs for joint bit error correction and packet error detection in HARQ systems, and compares the throughput and complexity of TPC-based HARQ to conventional HARQ systems.

Most modern communication systems such as LTE [109] and WiMAX [110] employ a combination of FEC and ARQ to provide the required QoS for various applications. When ARQ is combined with FEC codes, the composite system is commonly referred to as hybrid ARQ (HARQ). In conventional HARQ systems, the error correction is performed in two phases. In the first phase, the FEC codes are applied to correct the bit errors in the received packets. In the second

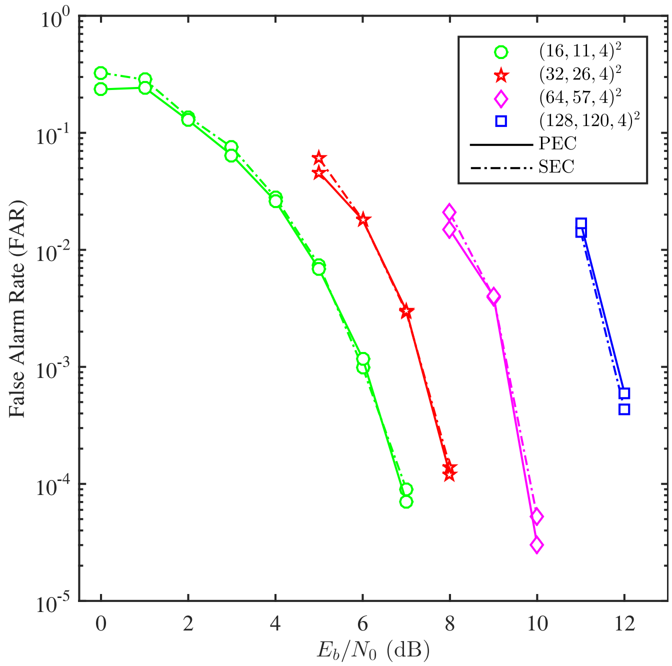

Fig. 9. False alarm rate of SEC-based and PEC-based self-detection.

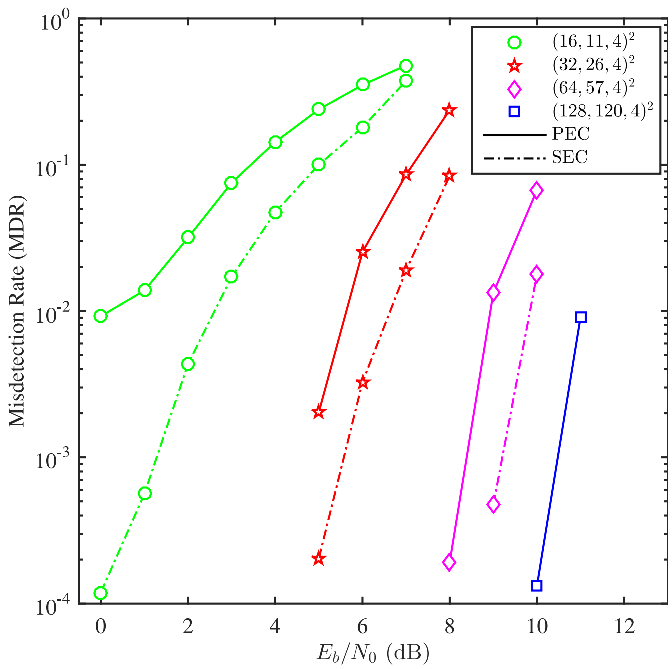

Fig. 10. Misdetection rate of SEC-based and PEC-based self-detection.

phase, the receiver verifies if the FEC process was successful, and then sends a message to the transmitter to retransmit the packet if the FEC process was unsuccessful; otherwise, the message informs the transmitter to send a new packet. In general, most of the work reported in the literature performs the bit error correction and packet error detection (PED) as two independent processes where the FEC is implemented at the Physical Layer (PHY) while the packet error detection is performed at the Data Link Control (DLC) layer [51], [111].

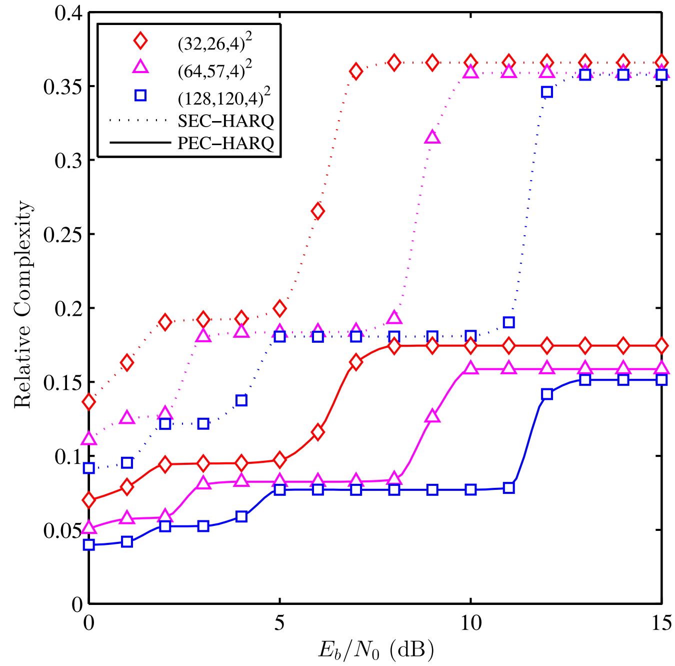

Fig. 11. Simulated relative complexity of TPCs PEC and SEC with respect to 16-bit CRC 8005 for HARQ with maximum number of allowed transmission rounds $L = 4$.

A. CRC-Free HARQ Using TPCs

TPCs error self-detection can be used to provide CRC-free PED to enhance the performance of HARQ systems in terms of complexity and throughput [22], [108], [112]. Complexity analysis of TPCs self-detection using SEC and PEC methods in Section IV-C demonstrates that a significant complexity reduction can be achieved using the CRC-free PED. Fig. 11 shows the relative complexity of TPCs self-detection schemes computed with respect to 16-bit CRC 8005 for a range of SNR values. The PEC method has the lowest complexity as compared to the SEC and CRC-based detection. The relative complexity of both SEC and PEC decreases as the SNR decreases. That is because the number of bit errors becomes higher at low SNR and an error is more likely to be detected at the beginning of the TPC codeword. The figure also shows that the relative complexity decreases as the codeword size increases.

The results are obtained from simulating an HARQ system with different eBCH product codes. The coded bits are modulated using BPSK and transmitted over independent and identically distributed (iid) Rayleigh fading with AWGN. The maximum number of ARQ rounds per packet is $L = 4$. The TPC decoder is configured to perform a maximum of $\mathcal{I}_{\max} = 4$ iterations. Moreover, the number of reliability bits for the Chase-II algorithm is set to $p = 4$ in the SISO decoder.

In addition to the complexity reduction, the TPCs self-detection systems offer throughput enhancement because the CRC bits can be replaced by information bits or FEC bits to support the error correction process. It is worth noting that the throughput enhancement is more apparent for short TPCs where the number of CRC bits is non-negligible compared

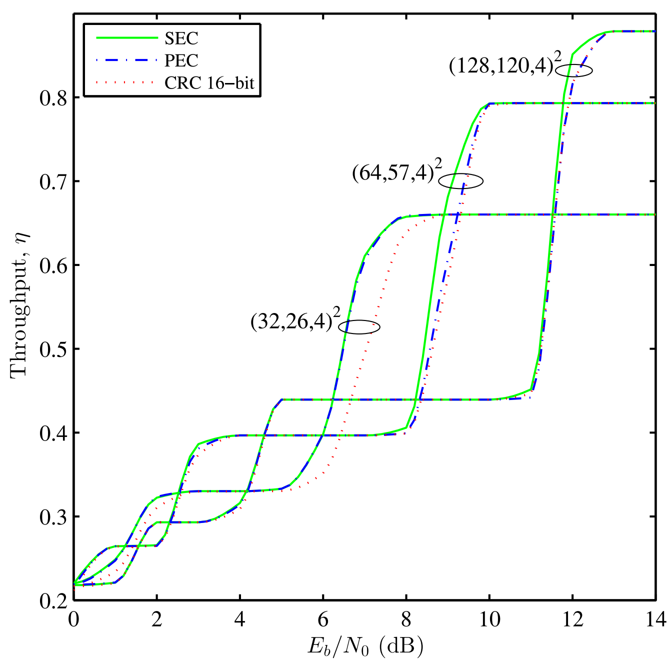

Fig. 12. Throughput of HARQ using PEC, SEC and 16-bit CRC with equal numbers of redundant bits.

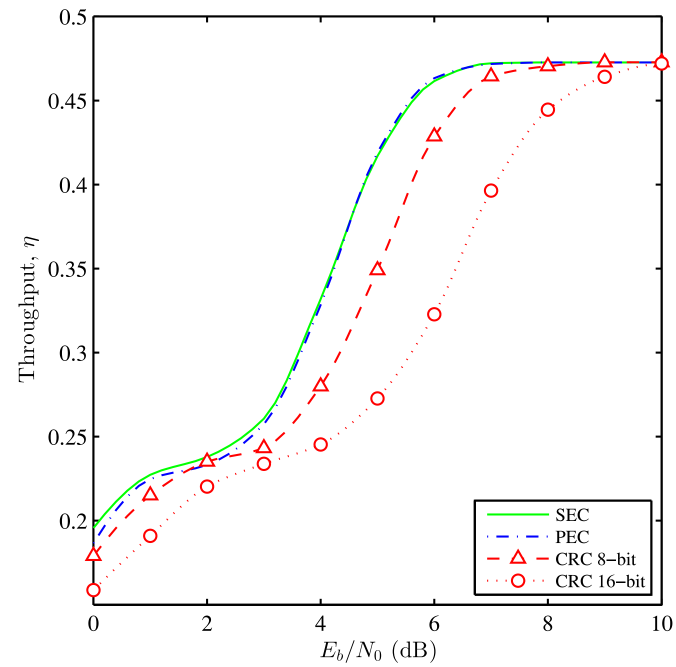

Fig. 13. Throughput of HARQ using PEC, SEC and CRC with 8 and 16 bits for eBCH$(16, 11, 4)^2$. All systems have equal number of redundant bits.

to the codeword size. The transmission efficiency or throughput η, is defined as the ratio of the number of information bits received successfully to the total number of transmitted bits [113, p. 461]. Assuming each transmitted packet is a TPC codeword, η is given by

$$\eta = \frac{1}{\mathbb{E}\{\rho\}} \frac{\kappa}{\mathcal{N}} (1 - P_{\mathrm{U}}) \quad (8)$$

where $\mathbb{E}\{\rho\}$ is the expected value of the number of transmissions per packet; $\frac{\kappa}{\mathcal{N}}$ is the code rate and P_{U} is the packet uncorrected error rate. It should be noted that a packet is considered to contain an uncorrected error if the first L transmission rounds fail, or if the packet contains an undetected error.

In practical HARQ systems, the error detection mechanism is not perfect where some correct packets are falsely rejected and some erroneous packets are misdetected. False rejections increase the number of retransmissions which degrades the system throughput. On the other hand, misdetections reduce the number of retransmissions and may cause throughput inflation if they are not accounted for when measuring or computing the throughput.

Fig. 12 compares the throughput of the CRC-based and CRC-free HARQ using equal code rates in all systems. In CRC systems, the redundant bits are composed of the CRC bits and the parity bits added by the TPC encoder. Therefore, to obtain equal code rates in CRC-based and CRC-free systems, the number of TPC parity bits should be reduced (punctured) by the number of CRC bit multiplied by the inverse of the code rate. The simulated throughput using equal code rates is depicted in Fig. 12 for relatively long TPCs. As it can be noted from the figure, SEC provides the highest throughput. PEC-HARQ provides higher throughput than CRC-based systems. However, the difference becomes very small as the TPC codeword size increases and almost vanishes at high SNRs. Such behavior is expected because the puncturing process reduces the error correction capability of the TPCs.

For short TPCs, the impact of the CRC bits length is more significant, and hence, smaller numbers of CRC bits can be employed. However, smaller number of CRC bits might result in unreliable performance because of the limited capability of such codes to detect the errors. Fig. 13 shows the throughput of HARQ using PEC, SEC and CRC with 8 and 16 bits for eBCH$(16, 11, 4)^2$. All systems have equal number of redundancy bits. As it can be noted from the figure, PEC and SEC have similar throughput for small codeword sizes and both have higher throughput than the CRC-based systems. The CRC-free and CRC-based systems converge to the same value at high SNRs. However, the puncturing process reduces the throughput of CRC-based systems at low SNR because their decoding process is less effective in terms of error correction. For the CRC-based systems, 8-bit CRC provides higher throughput than 16-bit CRC. However, although not shown in the figure, using a CRC code with less than 8 bits results in throughput degradation due to the high misdetection rate of the small CRC code.

B. TPCs-HARQ With Partial Retransmission

In [112], TPCs self-detection is employed to find the location of errors in the decoded codeword. For example, after the last column decoding process, row syndrome checking can be used to identify which rows are in error. Hence, partial retransmissions can be made instead of complete retransmissions of the codeword as in conventional HARQ systems. The TPCs row partial retransmission requires n additional feedback bits for each row. However, the performance gain in transmission efficiency due to a more selective retransmission can outweigh the negative impact of the additional feedback bits on the throughput.

C. Energy-Efficient TPCs-HARQ

The throughput of TPCs-HARQ varies in a staircase manner where it remains fixed for a wide range of SNR values as shown in Fig. 12. This unique behavior is also observed in other HARQ systems where capacity approaching FEC codes and packet combining are used [114]–[120]. The staircase behavior of the throughput implies that the transmit power can be controlled to minimize the total transmit energy while maintaining the throughput unchanged. In [120] and [121], it is shown that, for TPCs-HARQ, considerable power saving of up to 80% can be achieved without sacrificing the system throughput performance.

In addition, hardware performance analysis in [106] reveals that combining Hamming product codes with HARQ can result in 50% energy saving compared to other error control schemes.

D. Adaptive TPCs-HARQ

The throughput performance of adaptive TPCs-HARQ with iterative hard and soft decision decoding is considered in [51]. Monte Carlo simulation and semi-analytical solutions are used to evaluate the throughput of TPCs-HARQ for a wide selection of product codes. The obtained results reveal that the coding gain advantage of SISO over HIHO decoding is reduced significantly when the throughput is adopted as the performance metric, and it actually vanishes completely for some codes. When adaptive coding is used, the soft decoding advantage is limited to about 1.4 dB.

As discussed in Section III-C, TPCs SISO decoding requires considerable computational power compared to HIHO. The computational complexity constraint of TPCs becomes even more severe for systems that employ ARQ protocol because particular packets have to be retransmitted, and hence decoded several times.

In the literature, several techniques have been proposed to reduce the TPCs decoders complexity by improving the BER performance of the HIHO decoders [46], [70], [91]. Although such techniques managed to reduce the BER gap between SISO and HIHO decoders, the SISO BER remains considerably smaller. For example, the SISO eBCH$(32, 21, 6)^2$ and eBCH$(64, 51, 6)^2$ have a coding gain advantage of more than 2 dB over the HIHO ones in AWGN channels. The gap becomes larger with higher code rates as in the case of the eBCH$(32, 26, 4)^2$ and eBCH$(128, 120, 4)^2$, where the coding gain difference surges to about 4 dB [46]. Consequently, the low complexity might not be sufficient to justify adopting HIHO decoding for practical systems due to the high coding gain penalty.

In general, most of the work considered in the literature aimed at minimizing the computational complexity under fixed BER constraint [46], [75]. However, the BER is not necessarily sufficient to describe the QoS for systems that incorporate HARQ, where the throughput [117] or delay [122] are more desired performance metrics.

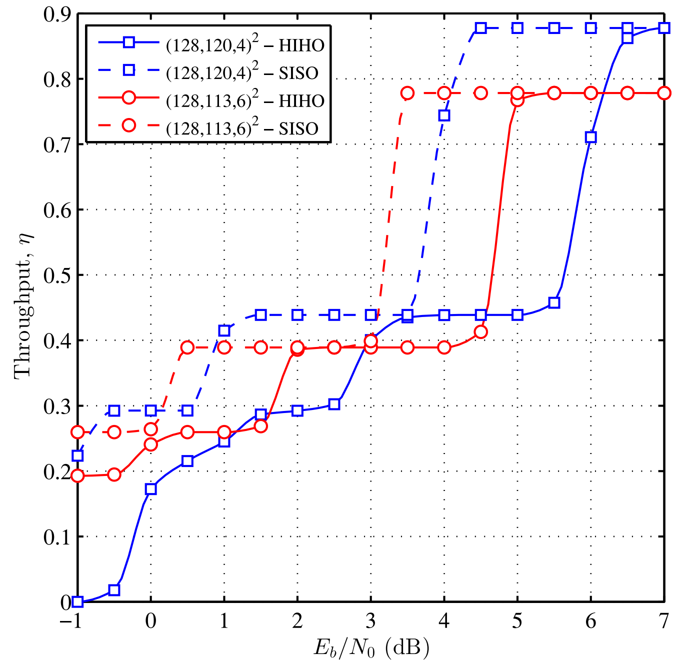

Fig. 14. Throughput of TPC-HARQ with packet combining for $(128, 120, 4)^2$ and $(128, 113, 6)^2$ in AWGN channels.

Fig. 14 shows the throughput results for the TPCs-HARQ with packet combining for eBCH $(128, 120, 4)^2$ and $(128, 113, 6)^2$ in AWGN channels. The throughput of the SISO and HIHO decoding for the $(128, 113, 6)^2$ is approximately equal for $E_b/N_0 \gtrsim 5$ dB, and in the range from 2 to 3 dB, which is remarkably different from the BER performance for these codes. The $(128, 120, 4)^2$ exhibits a similar behavior except that it is for a smaller range of E_b/N_0. This reveals that for particular codes and SNR values, significant complexity reduction can be achieved while providing the same throughput performance in TPCs-HARQ systems.

VI. Turbo Product Codes Challenges and Open Research Issues

Future research directions and open issues with regard to turbo product codes can be outlined as follows.

- Obtaining accurate theoretical models or exact expressions for the error performance of TPCs is still an open problem. Using Monte Carlo simulation to evaluate the performance of TPCs could be tedious, particularly for large values of $n_1 \times n_2$ at high SNRs. Moreover, theoretical models are helpful in understanding the characteristics of TPCs and their error correction and detection capabilities. Consequently, developing analytical solutions are indispensable for the evaluation and optimization of TPCs. Computing the bit error probability analytically is not feasible because TPCs error correction capability depends on the error pattern rather than the number of errors. Therefore, the performance of TPCs has been studied in the literature using bounds [55], [123]–[127].

However, to the best of our knowledge there are no closed form or exact expressions for the error probabilities of TPCs. Evaluating the throughput performance of TPCs-HARQ using Monte Carlo simulation could be even more tedious due to the retransmission process inherent in HARQ systems. However, a semi-analytical solution is proposed in [120] to obtain the throughput of HARQ systems in AWGN and iid Rayleigh fading channels from the packet error rate in one-shot transmission.

- A comprehensive study is needed to compare the performance of TPCs to other capacity-approaching codes such as PCCC, LDPC codes, and polar codes [128] over different channel models. The comparison should consider complexity analysis and error performance at different code rates. In the literature, the performance of TPCs is qualitatively compared to some codes for different applications. For example, according to [17] and [27], TPCs have lower decoding complexity and superior performance at very high code rates compared to PCCC. Similarly, [129] shows that TPCs have lower decoding latency and better error performance at high code rates compared to LDPC codes. Nevertheless, comprehensive evaluation and rigorous complexity analysis are still required to provide a comparison with a wide range of other competing codes.
- Improving the performance of TPCs in terms of complexity, error correction and self-detection is also an open research issue. In terms of complexity, the Chase-II decoder which is the core of the SISO decoding process represents the major portion of the decoder complexity. Designing a low-complexity alternative would have a significant impact on the overall decoder complexity. The work reported in the literature on this aspect is limited. A possible solution is to adjust the TPC decoder parameter p dynamically based on the channel SNR to provide the desired QoS with minimum complexity. Moreover, the TPCs row-column interleaver can be replaced by a random interleaver to enhance the TPCs error correction and detection of close-chain error patterns, particularly at low code rates and small codeword sizes.
- Combining TPCs with other error control and transmission techniques such as HARQ, space time block codes (STBC), orthogonal frequency division multiplexing (OFDM) and relay systems has recently received a lot of attention for the achieved high gains in energy, complexity, throughput and error performance [22], [51], [120], [130]–[132]. For example, a distributed TPC encoding is proposed in [132] where the source transmits eBCH encoded frames to a relay. The relay rearranges the decoded frames into rows and re-encodes them along the columns using a soft parity generation method. The performance of the proposed scheme is evaluated for few source to destination SNR values and for different positions of the relay along the line between source and destination. The obtained results show that the proposed scheme can outperform other schemes such as decode and forward and non-cooperative systems.

VII. Conclusion

This work presented a comprehensive study of TPCs, where state-of-the-art encoding and decoding techniques were classified and appraised. Based on the literature surveyed, it was noticed that TPCs are promising error correction techniques because they possess several desirable features in terms of error performance, flexibility of design, available code rates and code sizes. Moreover, TPCs exhibit unique throughput performance in HARQ systems which can be used effectively to optimize the transmission power. However, TPCs can be made more attractive if some limitations, such as SISO decoding computational complexity, are resolved.

The main parameters that should be considered for TPCs design and integration in communications systems are the code rate, codeword length and number of iterations, which affect the spectral efficiency, latency and computational complexity, respectively. The impact of these parameters, except for the number of iterations, on the error performance is not significant for several codes of interest where the coding gain difference between such codes is within 1 dB at BER of 10^{-5}. The number of iterations is the key parameter that determines the decoder complexity and error performance of TPCs. Although using four iterations is sufficient to provide most of the achievable coding gain, the computational complexity per iteration is considerable, and hence, the design of decoders with adaptive number of iterations is indispensable to reduce the decoder complexity. The spectral efficiency and latency have to be compromised based on the system requirements. While high spectral efficiency calls for using high code rates and high length codewords, the latency will increase as a consequence, and vice versa.

References

[1] M. A. Mehaseb, Y. Gadallah, A. Elhamy, and H. El-Hennawy, "Classification of LTE uplink scheduling techniques: An M2M perspective," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1310–1335, 2nd Quart. 2016.

[2] Y. Chen, T. Farley, and N. Ye, "QoS requirements of network applications on the Internet," *Inf. Knowl. Syst. Manag.*, vol. 4, no. 1, pp. 55–76, 2004.

[3] M. Alasti, B. Neekzad, J. Hui, and R. Vannithamby, "Quality of service in WiMAX and LTE networks [topics in wireless communications]," *IEEE Commun. Mag.*, vol. 48, no. 5, pp. 104–111, May 2010.

[4] A. Glavieux, *Channel Coding in Communication Networks: From Theory to Turbocodes*. London, U.K.: ISTE, 2007.

[5] P. Elias, "Error-free coding," *Trans. IRE Prof. Group Inf. Theory*, vol. 4, no. 4, pp. 29–37, Sep. 1954.

[6] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.

[7] R. M. Pyndiah, "Near-optimum decoding of product codes: Block turbo codes," *IEEE Trans. Commun.*, vol. 46, no. 8, pp. 1003–1010, Aug. 1998.

[8] J. Li, E. Kurtas, K. R. Narayanan, and C. N. Georghiades, "On the performance of turbo product codes over partial response channels," *IEEE Trans. Magn.*, vol. 37, no. 4, pp. 1932–1934, Jul. 2001.

[9] C. Argon and S. W. McLaughlin, "Optical OOK-CDMA and PPM-CDMA systems with turbo product codes," *J. Lightw. Technol.*, vol. 20, no. 9, pp. 1653–1663, Sep. 2002.

[10] L. Sun, H. Song, Z. Keirn, and B. V. K. V. Kumar, "Field programmable gate array (FPGA) for iterative code evaluation," *IEEE Trans. Magn.*, vol. 42, no. 2, pp. 226–231, Feb. 2006.

[11] C. Argon and S. W. McLaughlin, "A parallel decoder for low latency decoding of turbo product codes," *IEEE Commun. Lett.*, vol. 6, no. 2, pp. 70–72, Feb. 2002.

[12] J. Cuevas, P. Adde, S. Kerouedan, and R. Pyndiah, "New architecture for high data rate turbo decoding of product codes," in *Proc. IEEE Glob. Telecommun. Conf. (GLOBECOM)*, vol. 2. Taipei, Taiwan, Nov. 2002, pp. 1363–1367.

[13] C. Jego and P. Adde, "Row-column parallel turbo decoding of product codes," *Electron. Lett.*, vol. 42, no. 5, pp. 296–298, Mar. 2006.

[14] C. Jego, P. Adde, and C. Leroux, "Full-parallel architecture for turbo decoding of product codes," *Electron. Lett.*, vol. 42, no. 18, pp. 1052–1053, Aug. 2006.

[15] C. Leroux, C. Jego, P. Adde, M. Jezequel, and D. Gupta, "A highly parallel turbo product code decoder without interleaving resource," in *Proc. IEEE Workshop Signal Process. Syst. (SiPS)*, Washington, DC, USA, Oct. 2008, pp. 1–6.

[16] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE Int. Conf. Commun. (ICC)*, vol. 2. Geneva, Switzerland, May 1993, pp. 1064–1070.

[17] D. Minoli, *Satellite Systems Engineering in an IPv6 Environment*. Boca Raton, FL, USA: CRC Press, 2009.

[18] T. Hehn and J. B. Huber, "LDPC codes and convolutional codes with equal structural delay: A comparison," *IEEE Trans. Commun.*, vol. 57, no. 6, pp. 1683–1692, Jun. 2009.

[19] M. Kaiser, W. Fong, M. Sikora, and D. J. Costello, Jr., "A comparison of decoding latency for block and convolutional codes," in *Proc. Int. Symp. Commun. Theory Appl. (ISCTA)*, Ambleside, U.K., Jul. 2009, pp. 1–6.

[20] S. V. Maiya, D. J. Costello, and T. E. Fuja, "Low latency coding: Convolutional codes vs. LDPC codes," *IEEE Trans. Commun.*, vol. 60, no. 5, pp. 1215–1225, May 2012.

[21] C. Rachinger, R. Müller, and J. B. Huber, "Low latency-constrained high rate coding: LDPC codes vs. convolutional codes," in *Proc. Int. Symp. Turbo Codes Iterative Inf. Process. (ISTC)*, Bremen, Germany, Aug. 2014, pp. 218–222.

[22] H. Mukhtar, A. Al-Dweik, and M. Al-Mualla, "CRC-free hybrid ARQ system using turbo product codes," *IEEE Trans. Commun.*, vol. 62, no. 12, pp. 4220–4229, Dec. 2014.

[23] T. Mizuochi *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. Topics Quantum Electron.*, vol. 10, no. 2, pp. 376–386, Mar. 2004.

[24] T. Mizuochi, "Recent progress in forward error correction and its interplay with transmission impairments," *IEEE J. Sel. Topics Quantum Electron.*, vol. 12, no. 4, pp. 544–554, Jul./Aug. 2006.

[25] Y. Miyata *et al.*, "UEP-BCH product code based hard-decision FEC for 100 Gb/s optical transport networks," in *Proc. Opt. Fiber Commun. Conf. Expo. Nat. Fiber Opt. Eng. Conf. (OFC/NFOEC)*, Los Angeles, CA, USA, Mar. 2012, pp. 1–3.

[26] B. Li, K. J. Larsen, D. Zibar, and I. T. Monroy, "Reconfigurable forward error correction decoder for beyond 100 Gbps high speed optical links," *IEEE Commun. Lett.*, vol. 19, no. 2, pp. 119–122, Feb. 2015.

[27] K. Sripimanwat, *Turbo Code Applications*. Dordrecht, The Netherlands: Springer, 2005.

[28] *IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Broadband Wireless Access Systems*, IEEE Standard 802.16, May 2009.

[29] *IEEE Standard for Local and Metropolitan Area Networks Part 20: Air Interface for Mobile Broadband Wireless Access Systems Supporting Vehicular Mobilityphysical and Media Access Control Layer Specification*, IEEE Standard 802.20, Aug. 2008.

[30] *IEEE Standard for Broadband Over Power Line Networks: Medium Access Control and Physical Layer Specifications*, IEEE Standard 1901, Dec. 2010.

[31] G. Bosco, G. Montorsi, and S. Benedetto, "Soft decoding in optical systems," *IEEE Trans. Commun.*, vol. 51, no. 8, pp. 1258–1265, Aug. 2003.

[32] W. Xi and T. Adali, "Integrated MAP equalization and coding for optical-fiber-communication systems," *J. Lightw. Technol.*, vol. 24, no. 10, pp. 3663–3670, Oct. 2006.

[33] N. McSparron. (2001). *Next Generation Direct-to-Home Satellite Systems, Advanced Hardware Architectures*. [Online]. Available: http://www.aha.com/Uploads/Direct-To-Home.pdf

[34] U. Vilaipornsawai and M. R. Soleymani, "A novel turbo coding scheme for satellite ATM using Reed–Muller codes," *IEEE Trans. Commun.*, vol. 51, no. 5, pp. 767–773, May 2003.

[35] THAICOM Public Company Limited. (2009). *iCON Satellite Terminal, IPSTAR Broadband Satellite*. [Online]. Available: http://www.ipstar.com/pdf/products/iCON.pdf

[36] THAICOM Public Company Limited. (2009). *Enterprise Series, IPSTAR Broadband Satellite*. [Online]. Available: http://www.ipstar.com/pdf/products/enterprise.pdf

[37] L. Cao and C. W. Chen, "A novel product coding and recurrent alternate decoding scheme for image transmission over noisy channels," *IEEE Trans. Commun.*, vol. 51, no. 9, pp. 1426–1431, Sep. 2003.

[38] V. Janakiraman, J. S. Jang, and J. Kim, "Transmission of block turbo coded digital audio over FM-SCA," *IEEE Trans. Broadcast.*, vol. 50, no. 2, pp. 107–112, Jun. 2004.

[39] J. Poklemba and D. Wenzel, "Turbo-product-coded QVSB," *IEEE Trans. Broadcast.*, vol. 52, no. 4, pp. 579–584, Dec. 2006.

[40] Y. Sun, Z. Xiong, and X. Wang, "Iterative decoding of differentially space-time coded multiple descriptions of images," *IEEE Signal Process. Lett.*, vol. 11, no. 8, pp. 686–689, Aug. 2004.

[41] N. Thomos, N. V. Boulgouris, and M. G. Strintzis, "Wireless image transmission using turbo codes and optimal unequal error protection," *IEEE Trans. Image Process.*, vol. 14, no. 11, pp. 1890–1901, Nov. 2005.

[42] Y. Wang, Y. Du, S. Yu, and K. T. Chan, "An adaptive UEP_BTC_STBC_OFDM system for robust video transmission," in *Proc. IEEE Workshop Multimedia Signal Process.*, Shanghai, China, Oct./Nov. 2005, pp. 1–4.

[43] N. Thomos, N. V. Boulgouris, and M. G. Strintzis, "Optimized transmission of JPEG2000 streams over wireless channels," *IEEE Trans. Image Process.*, vol. 15, no. 1, pp. 54–67, Jan. 2006.

[44] L. Yao and L. Cao, "Turbo codes-based image transmission for channels with multiple types of distortion," *IEEE Trans. Image Process.*, vol. 17, no. 11, pp. 2112–2121, Nov. 2008.

[45] S. Yang, Y. Han, X. Wu, R. Wood, and R. Galbraith, "A soft decodable concatenated LDPC code," *IEEE Trans. Magn.*, vol. 51, no. 11, pp. 1–4, Nov. 2015.

[46] A. J. Al-Dweik and B. S. Sharif, "Closed-chains error correction technique for turbo product codes," *IEEE Trans. Commun.*, vol. 59, no. 3, pp. 632–638, Mar. 2011.

[47] C. Xu, Y.-C. Liang, and W. S. Leon, "Shortened turbo product codes: Encoding design and decoding algorithm," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3495–3501, Nov. 2007.

[48] C. Xu, Y.-C. Liang, and W. S. Leon, "A low complexity decoding algorithm for extended turbo product codes," *IEEE Trans. Wireless Commun.*, vol. 7, no. 1, pp. 43–47, Jan. 2008.

[49] D. Morero and M. Hueda, "Novel serial code concatenation strategies for error floor mitigation of low-density parity-check and turbo product codes," *Can. J. Elect. Comput. Eng.*, vol. 36, no. 2, pp. 52–59, 2013.

[50] S. Hirasawa, M. Kasahara, Y. Sugiyama, and T. Namekawa, "Modified product codes," *IEEE Trans. Inf. Theory*, vol. 30, no. 2, pp. 299–306, Mar. 1984.

[51] H. Mukhtar, A. Al-Dweik, M. Al-Mualla, and A. Shami, "Adaptive hybrid ARQ system using turbo product codes with hard/soft decoding," *IEEE Commun. Lett.*, vol. 17, no. 11, pp. 2132–2135, Nov. 2013.

[52] R. Zhou, R. Le Bidan, R. Pyndiah, and A. Goalic, "Low-complexity high-rate Reed–Solomon block turbo codes," *IEEE Trans. Commun.*, vol. 55, no. 9, pp. 1656–1660, Sep. 2007.

[53] B. Liu, Y. Li, B. Rong, L. Gui, and Y. Wu, "LDPC-RS product codes for digital terrestrial broadcasting transmission system," *IEEE Trans. Broadcast.*, vol. 60, no. 1, pp. 38–49, Mar. 2014.

[54] D. M. Rankin and T. A. Gulliver, "Single parity check product codes," *IEEE Trans. Commun.*, vol. 49, no. 8, pp. 1354–1362, Aug. 2001.

[55] D. M. Rankin, T. A. Gulliver, and D. P. Taylor, "Asymptotic performance of single parity-check product codes," *IEEE Trans. Inf. Theory*, vol. 49, no. 9, pp. 2230–2235, Sep. 2003.

[56] D.-W. Yue and E.-H. Yang, "Asymptotically Gaussian weight distribution and performance of multicomponent turbo block codes and product codes," *IEEE Trans. Commun.*, vol. 52, no. 5, pp. 728–736, May 2004.

[57] R. Pyndiah, A. Picart, and A. Glavieux, "Performance of block turbo coded 16-QAM and 64-QAM modulations," in *Proc. IEEE Glob. Telecommun. Conf. (GLOBECOM)*, vol. 2. Singapore, Nov. 1995, pp. 1039–1043.

[58] J. Li, K. R. Narayanan, and C. N. Georghiades, "Product accumulate codes: A class of codes with near-capacity performance and low decoding complexity," *IEEE Trans. Inf. Theory*, vol. 50, no. 1, pp. 31–46, Jan. 2004.

[59] J. P. Fonseka, E. M. Dowling, T. K. Brown, and S. I. Han, "Constrained interleaving of turbo product codes," *IEEE Commun. Lett.*, vol. 16, no. 9, pp. 1365–1368, Sep. 2012.

[60] D. J. Costello and G. D. Forney, "Channel coding: The road to channel capacity," *Proc. IEEE*, vol. 95, no. 6, pp. 1150–1177, Jun. 2007.

[61] M. Baldi, G. Cancellieri, and F. Chiaraluce, "Interleaved product LDPC codes," *IEEE Trans. Commun.*, vol. 60, no. 4, pp. 895–901, Apr. 2012.

[62] M. Alipour, O. Etesami, G. Maatouk, and A. Shokrollahi, "Irregular product codes," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Lausanne, Switzerland, Sep. 2012, pp. 197–201.

[63] J. A. Alzubi, O. A. Alzubi, and T. M. Chen, *Forward Error Correction Based on Algebraic-Geometric Theory*. Cham, Switzerland: Springer, 2014.

[64] O. Amrani, "Nonlinear codes: The product construction," *IEEE Trans. Commun.*, vol. 55, no. 10, pp. 1845–1851, Oct. 2007.

[65] J. G. Proakis and M. Salehi, *Digital Communications*, 5th ed. New York, NY, USA: McGraw-Hill, 2008.

[66] E. R. Berlekamp, *Algebraic Coding Theory*. New York, NY, USA: McGraw-Hill, 1968.

[67] J. Massey, "Shift-register synthesis and BCH decoding," *IEEE Trans. Inf. Theory*, vol. 15, no. 1, pp. 122–127, Jan. 1969.

[68] N. Abramson, "Cascade decoding of cyclic product codes," *IEEE Trans. Commun. Technol.*, vol. 16, no. 3, pp. 398–402, Jun. 1968.

[69] J. D. Andersen, "Product codes for optical communication," in *Proc. Eur. Conf. Opt. Commun. (ECOC)*, vol. 3. Copenhagen, Denmark, Sep. 2002, pp. 1–2.

[70] A. J. Al-Dweik and B. S. Sharif, "Non-sequential decoding algorithm for hard iterative turbo product codes," *IEEE Trans. Commun.*, vol. 57, no. 6, pp. 1545–1549, Jun. 2009.

[71] T. Janvars and P. Farkaš, "Hard decision decoding of single parity turbo product code with N-level quantization," in *Proc. Int. Conf. Telecommun. Signal Process. (TSP)*, Prague, Czech Republic, Jul. 2015, pp. 1–6.

[72] D. Chase, "Class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 170–182, Jan. 1972.

[73] A. J. Al-Dweik. (2016). *Turbo Product Code MATLAB Implementation*. [Online]. Available: https://sites.google.com/a/fulbrightmail.org/dweik/

[74] C. Jego and W. J. Gross, "Turbo decoding of product codes using adaptive belief propagation," *IEEE Trans. Commun.*, vol. 57, no. 10, pp. 2864–2867, Oct. 2009.

[75] S. Dave, J. Kim, and S. C. Kwatra, "An efficient decoding algorithm for block turbo codes," *IEEE Trans. Commun.*, vol. 49, no. 1, pp. 41–46, Jan. 2001.

[76] T. Kaneko, T. Nishijima, H. Inazumi, and S. Hirasawa, "An efficient maximum-likelihood-decoding algorithm for linear block codes with algebraic decoder," *IEEE Trans. Inf. Theory*, vol. 40, no. 2, pp. 320–327, Mar. 1994.

[77] M. P. C. Fossorier and S. Lin, "Soft-decision decoding of linear block codes based on ordered statistics," *IEEE Trans. Inf. Theory*, vol. 41, no. 5, pp. 1379–1396, Sep. 1995.

[78] P. A. Martin, D. P. Taylor, and M. P. C. Fossorier, "Soft-input soft-output list-based decoding algorithm," *IEEE Trans. Commun.*, vol. 52, no. 2, pp. 252–262, Feb. 2004.

[79] M. Loncar, R. Johannesson, I. E. Bocharova, and B. D. Kudryashov, "Soft-output BEAST decoding with application to product codes," *IEEE Trans. Inf. Theory*, vol. 54, no. 3, pp. 1036–1049, Mar. 2008.

[80] G. Liva, E. Paolini, and M. Chiani, "On optimum decoding of certain product codes," *IEEE Commun. Lett.*, vol. 18, no. 6, pp. 905–908, Jun. 2014.

[81] A. Zanko, A. Leshem, and E. Zehavi, "Analog product codes decodable by linear programming," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 509–518, Feb. 2012.

[82] Q. Zhang and T. Le-Ngoc, "Turbo product codes for FH-SS with partial-band interference," *IEEE Trans. Wireless Commun.*, vol. 1, no. 3, pp. 513–520, Jul. 2002.

[83] M. B. Pursley and J. S. Skinner, "Adaptive coding for frequency-hop transmission in mobile ad hoc networks with partial-band interference," *IEEE Trans. Commun.*, vol. 57, no. 3, pp. 801–811, Mar. 2009.

[84] M. R. Masse, M. B. Pursley, and J. S. Skinner, "Adaptive coding for frequency-hop transmission over fading channels with partial-band interference," *IEEE Trans. Commun.*, vol. 59, no. 3, pp. 854–862, Mar. 2011.

[85] C. Xu, Y.-C. Liang, Y. L. Guan, and W. S. Leon, "Turbo product codes for mobile multimedia broadcasting with partial-time jamming," *IEEE Trans. Broadcast.*, vol. 53, no. 1, pp. 256–262, Mar. 2007.

[86] J. Li, K. R. Narayanan, E. Kurtas, and C. N. Georghiades, "On the performance of high-rate TPC/SPC codes and LDPC codes over partial response channels," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 723–734, May 2002.

[87] B. G. Bajoga and W. J. Walbesser, "Decoder complexity for BCH codes," *Proc. IEE*, vol. 120, no. 4, pp. 429–431, Apr. 1973.

[88] S. A. Hirst, B. Honary, and G. Markarian, "Fast Chase algorithm with an application in turbo decoding," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1693–1699, Oct. 2001.

[89] C. Argon and S. W. McLaughlin, "An efficient Chase decoder for turbo product codes," *IEEE Trans. Commun.*, vol. 52, no. 6, pp. 896–898, Jun. 2004.

[90] G. T. Chen, L. Cao, L. Yu, and C. W. Chen, "Test-pattern-reduced decoding for turbo product codes with multi-error-correcting eBCH codes," *IEEE Trans. Commun.*, vol. 57, no. 2, pp. 307–310, Feb. 2009.

[91] A. Al-Dweik, S. Le Goff, and B. Sharif, "A hybrid decoder for block turbo codes," *IEEE Trans. Commun.*, vol. 57, no. 5, pp. 1229–1232, May 2009.

[92] P.-Y. Lu, E.-H. Lu, and T.-C. Chen, "An efficient hybrid decoder for block turbo codes," *IEEE Commun. Lett.*, vol. 18, no. 12, pp. 2077–2080, Dec. 2014.

[93] K. Ravid and O. Amrani, "MER: Mixed error reduction approach for soft-decision decoding of product block codes," *IEEE Trans. Commun.*, vol. 62, no. 9, pp. 3062–3069, Sep. 2014.

[94] C. Studer, C. Benkeser, S. Belfanti, and Q. Huang, "Design and implementation of a parallel turbo-decoder ASIC for 3GPP-LTE," *IEEE J. Solid-State Circuits*, vol. 46, no. 1, pp. 8–17, Jan. 2011.

[95] R. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[96] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electron. Lett.*, vol. 33, no. 6, pp. 457–458, Mar. 1997.

[97] F. Babich, G. Montorsi, and F. Vatta, "Some notes on rate-compatible punctured turbo codes (RCPTC) design," *IEEE Trans. Commun.*, vol. 52, no. 5, pp. 681–684, May 2004.

[98] C. Douillard and C. Berrou, "Turbo codes with rate-m/(m+1) constituent convolutional codes," *IEEE Trans. Commun.*, vol. 53, no. 10, pp. 1630–1638, Oct. 2005.

[99] M. Grymel and S. B. Furber, "A novel programmable parallel CRC circuit," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 10, pp. 1898–1902, Oct. 2011.

[100] G. D. Nguyen, "Fast CRCs," *IEEE Trans. Comput.*, vol. 58, no. 10, pp. 1321–1331, Oct. 2009.

[101] M. Ayinala and K. K. Parhi, "High-speed parallel architectures for linear feedback shift registers," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4459–4469, Sep. 2011.

[102] J. Satran, D. Sheinwald, and I. Shimony, "Out of order incremental CRC computation," *IEEE Trans. Comput.*, vol. 54, no. 9, pp. 1178–1181, Sep. 2005.

[103] P. Coulton, C. Tanriover, B. Wright, and B. Honary, "Simple hybrid type II ARQ technique using soft output information," *Electron. Lett.*, vol. 36, no. 20, pp. 1716–1717, Sep. 2000.

[104] M. E. Buckley and S. B. Wicker, "The design and performance of a neural network for predicting turbo decoding error with application to hybrid ARQ protocols," *IEEE Trans. Commun.*, vol. 48, no. 4, pp. 566–576, Apr. 2000.

[105] F. Zhai and I. J. Fair, "Techniques for early stopping and error detection in turbo decoding," *IEEE Trans. Commun.*, vol. 51, no. 10, pp. 1617–1623, Oct. 2003.

[106] B. Fu and P. Ampadu, "On Hamming product codes with type-II hybrid ARQ for on-chip interconnects," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 56, no. 9, pp. 2042–2054, Sep. 2009.

[107] J. C. Fricke and P. A. Hoeher, "Reliability-based retransmission criteria for hybrid ARQ," *IEEE Trans. Commun.*, vol. 57, no. 8, pp. 2181–2184, Aug. 2009.

[108] H. Mukhtar, A. Al-Dweik, and M. Al-Mualla, "Low complexity hybrid ARQ using extended turbo product codes self-detection," in *Proc. IEEE Glob. Telecommun. Conf. (GLOBECOM)*, San Diego, CA, USA, Dec. 2015, pp. 1–6.

[109] *Evolved Universal Terrestrial Radio Access (E-UTRA)*, 3GPP Standard TS 36.201 V11.0.0, Oct. 2012.

[110] *IEEE Standard for Wireless MAN-Advanced Air Interface for Broadband Wireless Access Systems*, IEEE Standard 802.16.1-2012, Sep. 2012.

[111] J. Ramis and G. Femenias, "Cross-layer design of adaptive multirate wireless networks using truncated HARQ," *IEEE Trans. Veh. Technol.*, vol. 60, no. 3, pp. 944–954, Mar. 2011.

[112] H. Mukhtar, A. Al-Dweik, and M. Al-Mualla, "Hybrid ARQ with partial retransmission using turbo product codes," in *Proc. Int. Conf. Inf. Commun. Technol. Res. (ICTRC)*, May 2015, pp. 28–31.

[113] S. Lin and D. J. Costello, *Error Control Coding*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1983.

[114] S. Kallel and D. Haccoun, “Generalized type II hybrid ARQ scheme using punctured convolutional coding,” *IEEE Trans. Commun.*, vol. 38, no. 11, pp. 1938–1946, Nov. 1990.
[115] W. Yafeng, Z. Lei, and Y. Dacheng, “Performance analysis of type III HARQ with turbo codes,” in *Proc. 57th IEEE Veh. Technol. Conf. (VTC)*, vol. 4. Apr. 2003, pp. 2740–2744.
[116] Q. Huang, S. Chan, L. Ping, and M. Zukerman, “Improving wireless TCP throughput by a novel TCM-based hybrid ARQ,” *IEEE Trans. Wireless Commun.*, vol. 6, no. 7, pp. 2476–2485, Jul. 2007.
[117] T.-Y. Lin, S.-K. Lee, H.-H. Tang, and M.-C. Lin, “An adaptive hybrid ARQ scheme with constant packet lengths,” *IEEE Trans. Commun.*, vol. 60, no. 10, pp. 2829–2840, Oct. 2012.
[118] C. J. Le Martret, A. Le Duc, S. Marcille, and P. Ciblat, “Analytical performance derivation of hybrid ARQ schemes at IP layer,” *IEEE Trans. Commun.*, vol. 60, no. 5, pp. 1305–1314, May 2012.
[119] M. El Aoun, X. Lagrange, R. Le Bidan, and R. Pyndiah, “Throughput analysis of hybrid single-packet and multiple-packet truncated type-II HARQ strategies with unreliable feedback channel,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Shanghai, China, Apr. 2012, pp. 86–91.
[120] H. Mukhtar, A. Al-Dweik, M. Al-Mualla, and A. Shami, “Low complexity power optimization algorithm for multimedia transmission over wireless networks,” *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 1, pp. 113–124, Feb. 2015.
[121] H. Mukhtar, A. Al-Dweik, and M. Al-Mualla, “On the performance of adaptive HARQ with no channel state information feedback,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, New Orleans, LA, USA, Mar. 2015, pp. 446–451.
[122] N. Gunaseelan, L. Liu, J.-F. Chamberland, and G. H. Huff, “Performance analysis of wireless hybrid-ARQ systems with delay-sensitive traffic,” *IEEE Trans. Commun.*, vol. 58, no. 4, pp. 1262–1272, Apr. 2010.
[123] A. Sella and Y. Beery, “Convergence analysis of turbo decoding of product codes,” *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 723–735, Feb. 2001.
[124] F. Lehmann and G. M. Maggio, “Analysis of the iterative decoding of LDPC and product codes using the Gaussian approximation,” *IEEE Trans. Inf. Theory*, vol. 49, no. 11, pp. 2993–3000, Nov. 2003.
[125] F. Chiaraluce and R. Garello, “Extended Hamming product codes analytical performance evaluation for low error rate applications,” *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 2353–2361, Nov. 2004.
[126] M. Ferrari and S. Bellini, “Refinements and asymptotic performance of bandwidth-efficient turbo product codes,” *IEEE Trans. Commun.*, vol. 52, no. 6, pp. 866–870, Jun. 2004.
[127] J. Justesen, “Performance of product codes and related structures with iterated decoding,” *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 407–415, Feb. 2011.
[128] E. Arikan, “Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels,” *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.
[129] Comtech EF Data. (2010). *CDM-600/600L Open Network Satellite Modem*. [Online]. Available: http://www.comtechefdata.com/files/manuals/mn-modems-pdf/mn-cdm600-600L.pdf
[130] Y. Du and K. T. Chan, “Enhanced space-time block coded systems by concatenating turbo product codes,” *IEEE Commun. Lett.*, vol. 8, no. 6, pp. 388–390, Jun. 2004.
[131] M. Sabbaghian, Y. Kwak, B. Smida, and V. Tarokh, “Near Shannon limit and low peak to average power ratio turbo block coded OFDM,” *IEEE Trans. Commun.*, vol. 59, no. 8, pp. 2042–2045, Aug. 2011.
[132] E. A. Obiedat and L. Cao, “Soft information relaying for distributed turbo product codes (SIR-DTPC),” *IEEE Signal Process. Lett.*, vol. 17, no. 4, pp. 363–366, Apr. 2010.

H. Mukhtar (S’08–M’15) received the B.Sc. and M.Sc. degrees in electrical engineering from the American University of Sharjah, Sharjah, UAE, in 2004 and 2010, respectively, and the Ph.D. degree in electrical and computer engineering from Khalifa University, Abu Dhabi, UAE, in 2014, where he was as a Graduate Teaching Assistant with the Department of Electrical and Computer Engineering. From 2005 to 2008, he was a Technical Support and Project Engineer in the field of electronic security and surveillance in Dubai, UAE. He is currently a Post-Doctoral Research Fellow with the Visual Signal Analysis and Processing Research Center, Khalifa University. His current research interests include wireless communications, error control coding, and image/video analysis and processing.

A. Al-Dweik (S’9–M’01–SM’04) received the M.S. and Ph.D. degrees in electrical engineering from Cleveland State University, Cleveland, OH, USA, in 1998 and 2001, respectively. From 2003 to 2013, he was an Assistant and then an Associate Professor with the Department of Electrical and Computer Engineering, Khalifa University, UAE. He then joined the School of Engineering, University of Guelph, Guelph, ON, Canada, and he holds an Adjunct Research Professor position with Western University, London, ON, Canada. He has several years of industrial experience in the United States. His main research interests include wireless communications, synchronization techniques, OFDM technology, modeling and simulation of communication systems, error control coding, and spread spectrum systems. He was a recipient of the Fulbright Scholarship and several awards and research grants. He is an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.

A. Shami (M’03–SM’09) received the B.E. degree in electrical and computer engineering from Lebanese University, Beirut, Lebanon, in 1997, and the Ph.D. degree in electrical engineering from the Graduate School and University Center, City University of New York, New York, NY, USA. Since 2004, he has been with Western University, Canada, where he is currently a Professor with the Department of Electrical and Computer Engineering. His current research interests are in the area of network-based cloud computing and wireless/data networking. He is an Elected Chair of the IEEE Communications Society Technical Committee on Communications Software.