Iterative Decoding of LDPC Codes over the q-ary Partial Erasure Channel

Rami Cohen, *Graduate Student Member, IEEE,* and Yuval Cassuto, *Senior Member, IEEE*

***Abstract*—In this paper, we develop a new channel model, which we name the q-ary partial erasure channel (QPEC). The QPEC has a q-ary input, and its output is either the input symbol or a set of M ($2 \leq M \leq q$) symbols, containing the input symbol. This channel serves as a generalization to the binary erasure channel, and mimics situations when a symbol output from the channel is known only partially; that is, the output symbol contains some ambiguity, but is not fully erased. This type of channel is motivated by non-volatile memory multi-level read channels. In such channels the readout is obtained by a sequence of current/voltage measurements, which may terminate with partial knowledge of the stored level. Our investigation is concentrated on the performance of low-density parity-check (LDPC) codes when used over this channel, thanks to their low decoding complexity using belief propagation. We provide the exact QPEC density-evolution equations that govern the decoding process, and suggest a cardinality-based approximation as a proxy. We then provide several bounds and approximations on the proxy density evolutions, and verify their tightness through numerical experiments. Finally, we provide tools for the practical design of LDPC codes for use over the QPEC.**

***Index Terms*—Density evolution, belief propagation, low-density parity-check (LDPC) codes, non-volatile memories, q-ary codes, partial erasure, iterative decoding, decoding threshold, erasure channels.**

I. Introduction

The rapid development of memory technologies have introduced challenges to the continued scaling of memory devices in density and access speed. One of the common computer memory technologies is non-volatile memory (NVM). In multi-level NVMs, such as flash memories, an information symbol is represented in a memory cell by one of q voltage levels, where information is written/read by adding/measuring cell voltage [1], [2]. The read process is usually performed by comparing the stored voltage level to certain threshold voltage levels. To scale storage density in NVMs, the number of levels per cell is continuously increased [3], [4]. As the number of levels increases, errors become more and more prevalent due to intercell interference [5]. In addition, the use of multi-level memory cells in the emerging technology of resistive memories introduces significant reliability challenges [6].

Apart from classical channels and error models, multi-level memories motivate coding for a diversity of new channels with rich features. Our work here is motivated by a class of channels we call *measurement channels*, in which information is written/read by adding/measuring electrical charges. These channels encompass a variety of equivocations introduced to the information by an imperfect read process, due to either physical limitations or speed constraints. The channel we propose and study here – the *q-ary partial erasure channel* (QPEC) – is a basic and natural model for a measurement channel in multi-level memories. The model comes from a read process that occasionally fails to read the information at its entirety, and provides as decoder inputs q-ary symbols that are *partially* erased. In the QPEC model, an output symbol is a *set* of symbols that includes the correct input symbol. This set can be either of cardinality 1 or a set of M symbols (M is a parameter, $2 \leq M \leq q$). In the latter case, we say that a *partial erasure* event occurred, modeling the uncertainty that may occur in the read process due to imperfect measurements. Theoretically speaking, the QPEC is a generalization of the binary (or q-ary) erasure channel (BEC or QEC). In the BEC/QEC, symbols are either received perfectly or erased completely; in the QPEC, partially erased symbol provide information in quantity that grows as M gets smaller.

In this work, we suggest the use of GF(q) low-density parity-check (LDPC) codes [7]–[9] for encoding information over the QPEC, due to their low complexity of implementation and good performance under iterative decoding [10]. These codes were shown to achieve performance close to the capacity for several important channels, using efficient decoding algorithms [10], [11]. Non-binary LDPC codes were considered in several works, such as in [9], [12]–[14], and were shown as superior to binary codes in several cases [9], [14]. In our analysis, we propose a message-passing decoder for decoding GF(q)-LDPC codes over the QPEC, extending the known iterative decoder for the BEC/QEC to deal with partial erasures. The iterative-decoding performance evaluation of LDPC codes is usually performed using the density-evolution method [15] that tracks the decoding failure probability. However, this method becomes prohibitively complex in practice as q increases, as it requires an iterative evaluation of multi-dimensional probability distributions [13], [15]. Thus, we provide approximation schemes for tracking the QPEC decoding performance efficiently and verify their tightness. Finally, we develop tools for the design of good LDPC codes for the QPEC.

The authors are with the Department of Electrical Engineering, Technion - Israel Institute of Technology, Haifa, Israel 3200003 (email: rc@campus.technion.ac.il, ycassuto@ee.technion.ac.il)

This work was supported in part by the German-Israel Foundation, by the Israel Ministry of Science and Technology, and by the Israel Science Foundation.

Parts of this work were presented at the 2014 IEEE International Symposium on Information Theory (ISIT), Hawaii, USA, at the 8th International Symposium on Turbo Codes & Iterative Information Processing (ISTC), Bremen, Germany, and at the 2015 IEEE International Workshop on Information Theory (ITW), Jerusalem, Israel.

The paper is structured as follows. We begin by introducing the QPEC channel and its capacity in Section II. In Section III, we give a short review of GF(q)-LDPC codes and propose a message-passing based decoder for the QPEC. Decoding performance analysis is provided in Sections IV and V, and code design tools are discussed in Section VI. Finally, conclusions are given in Section VII.

II. The q-ary Partial Erasure Channel

A. Channel model

The *q-ary partial erasure channel* (QPEC) is a generalization of the well known binary erasure channel (BEC) [16] in two ways. First, similarly to the q-ary erasure channel (QEC), its input alphabet is q-ary, with $q \geq 2$. Second, generalizing the BEC erasure event, a partial erasure occurs when the input symbol is known to belong a set of M (M is a parameter, $2 \leq M \leq q$) symbols (rather than q symbols). The QPEC is defined as follows. Let X be the transmitted symbol, taken from the alphabet $\mathcal{X} = \left\{0, \alpha^0 = 1, \alpha^1, ..., \alpha^{q-2}\right\}$ of cardinality q, where α is a primitive element of the finite field GF(q) (i.e., the elements in $\mathcal{X}$ are the field elements). Define the super-symbols $?_x^{(i)}$ for each $x \in \mathcal{X}$ and for $i = 1, 2, ..., \binom{q-1}{M-1}$, such that $?_x^{(i)}$ is a set of M symbols containing the symbol x and $M-1$ other symbols, taken from $\mathcal{X} \backslash \{x\}$. The output Y (given an input symbol x) is a *set* of symbols, which is either the singleton $\{x\}$, or one of the sets $?_x^{(i)}$ (of cardinality M) for some i. Therefore, the output alphabet $\mathcal{Y}$ contains all possible sets of cardinality 1 and cardinality M taken from $\mathcal{X}$. The transition probabilities governing the QPEC are as follows:

$$\Pr\left(Y = y \middle| X = x\right) = \left\{ \begin{array}{cc} 1-\varepsilon, & y = \{x\} \\ \varepsilon / \binom{q-1}{M-1}, & y = ?_x^{(i)}, \end{array} \right. \tag{1}$$

where $0 \leq \varepsilon \leq 1$ is the (partial) erasure probability.

That is, with probability $1-\varepsilon$ the input symbol is received with no error, and with probability ε a partial-erasure event occurs, such that the input symbol is known up to M symbols. In the latter case, the output sets $?_x^{(i)}$ are equiprobable. This models a situation of maximum uncertainty at the output, which is uniformly distributed on sets of cardinality M containing x. Note that for $M = q = 2$ the QPEC is equivalent to the BEC, where for $M = q > 2$ the QPEC is equivalent to the QEC. The transition probabilities are given explicitly in the following example for a particular choice of q, M and a transmitted symbol x.

Example 1: Assume that $q = 4$ and that the symbol 0 was transmitted. If $M = 2$, the possible output sets, and their transition probabilities from (1), are given by:

$$\Pr\left(Y = y \middle| X = 0\right)_{q=4, M=2} = \left\{ \begin{array}{cc} 1-\varepsilon, & y = \{0\} \\ \varepsilon/3, & y = \left\{0, \alpha^0\right\} \\ \varepsilon/3, & y = \left\{0, \alpha^1\right\} \\ \varepsilon/3, & y = \left\{0, \alpha^2\right\}. \end{array} \right. \tag{2}$$

B. Capacity

Denote $p_x \triangleq \Pr\left(X = x\right)$, for $x = 0, \alpha^0, \alpha^1 ..., \alpha^{q-2}$, to be the input distribution to the channel. According to the definition of the channel capacity C,

$$C = \max_{\{p_x\}} I\left(X; Y\right) = \max_{\{p_x\}} \left(H\left(Y\right) - H\left(Y \middle| X\right)\right), \tag{3}$$

where $I\left(X; Y\right)$ is the mutual information between the input X and the output Y, and $H\left(Y\right)$, $H\left(Y \middle| X\right)$ are the entropy of Y and the conditional entropy of Y given X, respectively. The conditional entropy $H\left(Y \middle| X\right)$ can be calculated using (1):

$$H\left(Y \middle| X\right) = -\left(1-\varepsilon\right)\log\left(1-\varepsilon\right) - \varepsilon \log\left(\varepsilon / \binom{q-1}{M-1}\right). \tag{4}$$

The conditional entropy is independent of $\{p_x\}$ (as expected), implying that it is sufficient to maximize the entropy $H\left(Y\right)$ to find the capacity. The QPEC capacity is provided in the following theorem.

Theorem 1: (Capacity) The QPEC capacity is:

$$C\left(\text{QPEC}\right) = 1 - \varepsilon \log_q M, \tag{5}$$

measured in q-ary symbols per channel use.
The proof of this theorem is provided in Appendix A. As one may expect due to the uniform distribution of the output when a partial-erasure occurs, $H\left(Y\right)$ is maximized under the uniform distribution of the input (i.e., for $p_x = 1/q$). Note the agreement of (5) with the QEC capacity for $M = q$, and in particular with the BEC capacity for $M = q = 2$.

C. Maximum-likelihood decoding

Assume that a codeword $\boldsymbol{c}$ taken from a codebook $\mathcal{C}$ was transmitted over the QPEC and that the output $\boldsymbol{y}$ was received. The elements y_i of $\boldsymbol{y}$ should be understood in a generalized sense, as they contain either a set of one symbol or a set of M symbols (according to the transition probabilities in Equation (1)). For $M = q$, in which the QPEC is essentially the QEC, codewords coinciding with $\boldsymbol{y}$ in non-erased positions are said to be *compatible* with $\boldsymbol{y}$ [10], and they serve as maximum-likelihood (ML) decoding of $\boldsymbol{y}$. However, when $M < q$, partially-erased codeword symbol positions should be considered for the ML decoding of $\boldsymbol{y}$.

To extend the notion of compatibility to QPECs with $M < q$, we define the set:

$$\Psi = \left\{\boldsymbol{c} \in \mathcal{C} : \forall i, c_i \bigcap y_i \neq \emptyset\right\}, \tag{6}$$

which is the set of all codewords that have in each position a symbol that is contained in the corresponding output of $\boldsymbol{y}$ in the same position. Each codeword in Ψ can serve as an ML decoding of $\boldsymbol{c}$, since $\boldsymbol{c}$ and $\boldsymbol{y}$ must agree in non-erased positions, and in the remaining positions the correct transmitted codeword symbol c_i is contained in y_i by the QPEC definition. Therefore, $\boldsymbol{y}$ is decoded correctly (with probability 1) if and only if $|\Psi| = 1$. In a similar manner, when ML *symbol* decoding is used, y_i is decoded correctly (with probability 1) if and only if all the codewords in Ψ contain the same symbol in their i^{th} position. In practice, ML decoding complexity is usually prohibitive. In the next section, we move

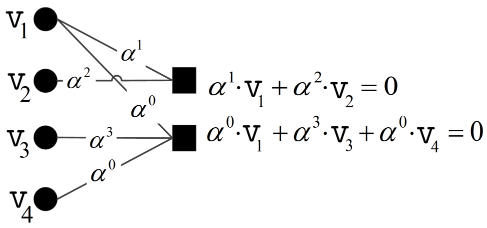


Fig. 1: An example of a Tanner graph over GF(4). Circles denote variable nodes (codeword symbols), and squares denote check nodes (parity-check equations). The symbols on the edges are the labels, leading to the parity-check equations on the right.

to specify a low-complexity iterative message-passing decoder for GF(q) LDPC codes used over the QPEC.

III. GF(q) LDPC Codes and Message-Passing Decoding

A. GF(q) LDPC codes

Before developing our coding results for the QPEC, we include some well-known facts on LDPC codes as a necessary background. A GF(q) $[n,k]$ LDPC code is defined in a similar way to its binary counterpart, by a sparse parity-check matrix, or equivalently by a Tanner graph [17]. This graph is bipartite, with n variable (left) nodes, which correspond to codeword symbols, and $n-k$ check (right) nodes, which correspond to parity-check equations. The codeword symbols are taken from GF(q), where the labels on the graph edges are taken from the non-zero elements of GF(q). In the graph, a check node $\mathtt{c}$ is connected by edges to variable nodes $\mathtt{v} \in \mathcal{N}(\mathtt{c})$, where $\mathcal{N}(\mathtt{c})$ denotes the set of variable nodes adjacent to check node $\mathtt{c}$. The induced parity-check equation is $\sum\limits_{\mathtt{v}\in N(\mathtt{c})} h_{\mathtt{c},\mathtt{v}} \cdot \mathtt{v} = 0$, where $h_{\mathtt{c},\mathtt{v}}$ are the labels on the edges connecting variable node $\mathtt{v}$ to check node $\mathtt{c}$. Note that the calculations are performed using GF(q) arithmetic. An example of a Tanner graph is given in Figure 1.

LDPC codes are usually characterized by the *degree distributions* of the variable nodes and the check nodes. They are called *regular* if both variable nodes and check nodes have constant degree. Otherwise, they are called *irregular*. Denote by d_v and d_c the maximal degree of variable nodes and check nodes, respectively. As is customary [10], we define the following degree-distribution polynomials:

$$\lambda\left(x\right) = \sum_{i=2}^{d_v} \lambda_i x^{i-1}, \tag{7}$$

$$\rho\left(x\right) = \sum_{i=2}^{d_c} \rho_i x^{i-1}, \tag{8}$$

where for each i, a fraction λ_i (ρ_i) of the edges is connected to variable (check) nodes of degree i. These polynomials will be used later for analyzing the iterative-decoding performance of LDPC codes over the QPEC. The *design rate* r of an LDPC code with degree-distribution polynomials $\lambda(x)$ and $\rho(x)$, measured in q-ary symbols per channel use, is [10]:

$$r = 1 - \frac{\int\limits_0^1 \rho\left(x\right)dx}{\int\limits_0^1 \lambda\left(x\right)dx} = 1 - \frac{\sum\limits_{i=2}^{d_c} \rho_i/i}{\sum\limits_{i=2}^{d_v} \lambda_i/i}. \tag{9}$$

The design rate equals to the actual rate if the rows of the LDPC code parity-check matrix are linearly independent.

B. Message-passing decoder for the QPEC

The following decoder for GF(q) LDPC codes over the QPEC is a variation of the standard message passing/belief propagation algorithm over a Tanner graph, generalizing the iterative decoding process used over the BEC/QEC. The key change is that in the QPEC setting the exchanged beliefs are sets of symbols, rather than individual symbols (and erasure symbols) as with the BEC/QEC. We have two types of messages at each decoding iteration l: *variable to check* (VTC) messages and *check to variable* (CTV) messages, denoted $\mathrm{VTC}^{(l)}_{\mathtt{v}\to\mathtt{c}}$ and $\mathrm{CTV}^{(l)}_{\mathtt{c}\to\mathtt{v}}$, respectively. Each outgoing message from a variable (check) node to a check (variable) node depends on all its incoming messages, except for the incoming message originated from the target node. At iteration $l = 0$, channel information is sent from variable nodes to check nodes: partially-erased nodes send sets of symbols of cardinality M, while non-erased ones send sets of cardinality 1 (recall that both sets contain the correct symbol). The channel information sent from variable node $\mathtt{v}$ will be denoted $\mathrm{VTC}^{(0)}_{\mathtt{v}}$.

In the subsequent iterations, the operations of the message-passing decoder translate to the following operations on sets. $\mathrm{CTV}^{(l)}_{\mathtt{c}\to\mathtt{v}}$ contains the possible values of $\mathtt{v}$ given the incoming messages from the variable nodes $\{\mathcal{N}\left(\mathtt{c}\right)\backslash\mathtt{v}\}$, such that the parity-check equation induced by check node $\mathtt{c}$ is satisfied. For later use, we note that the calculation of $\mathrm{CTV}^{(l)}_{\mathtt{c}\to\mathtt{v}}$ can be represented compactly, as follows. Define the *sumset* (or *Minkowski sum*) [18] operation between sets $\mathcal{S}_j$ $(j = 1, 2, ..., J)$ that contain GF(q) elements:

$$\sum_{j=1}^{J} \mathcal{S}_j = \left\{ \sum_{j=1}^{J} \mathcal{S}_j : s_j \in \mathcal{S}_j \right\}. \tag{10}$$

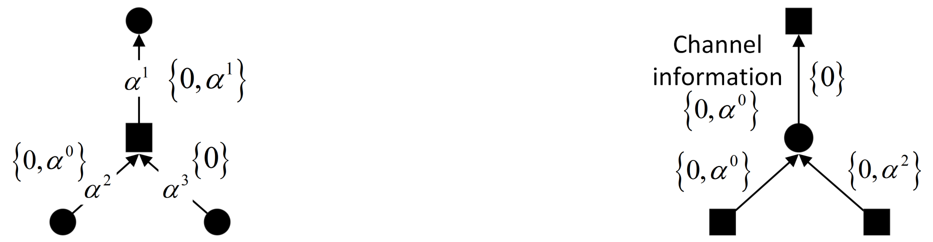
That is, the sumset results in a set containing all sums (using GF(q) arithmetic) of elements taken from $\mathcal{S}_j$.

Example 2: Assume that $q = 4$ and consider the sets $\left\{0, \alpha^0\right\}$ and $\left\{0, \alpha^1\right\}$. The sumset of these sets is $\left\{0, \alpha^0\right\} + \left\{0, \alpha^1\right\} = \left\{0+0, 0+\alpha^1, \alpha^0+0, \alpha^0+\alpha^1\right\} = \left\{0, \alpha^0, \alpha^1, \alpha^2\right\}$, i.e., all the field elements. On the other hand, if both sets are $\left\{0, \alpha^0\right\}$, then the sumset is $\left\{0, \alpha^0\right\} + \left\{0, \alpha^0\right\} = \left\{0, \alpha^0\right\}$.

For each pair of check node $\mathtt{c}$ and variable node $\mathtt{v}$, define the following sets for each $\mathtt{v}' \in \{\mathcal{N}\left(\mathtt{c}\right)\backslash\mathtt{v}\}$:

$$\mathcal{B}^{(l)}_{\mathtt{v}'} = \left\{ -h_{\mathtt{c},\mathtt{v}'} \cdot g' : g' \in \mathrm{VTC}^{(l-1)}_{\mathtt{v}'\to\mathtt{c}} \right\}, \quad l \geq 1 \tag{11}$$

which are the VTC message sets sent from the variable nodes adjacent to $\mathtt{c}$ except $\mathtt{v}$ to check node $\mathtt{c}$ at iteration $l-1$, multiplied by the additive inverses of the edge labels. We can



(a) An example for a CTV message over GF(4). The outgoing message is $\frac{\alpha^2}{\alpha} \cdot \{0, \alpha^0\} + \frac{\alpha^3}{\alpha} \cdot \{0\} = \{0, \alpha^1\}$.

(b) An example for a VTC message. The edge labels are not shown as they are not required for the VTC message calculation.

Fig. 2: Examples for CTV/VTC messages in the decoding process. Circles denote variable nodes, and squares denote check nodes. Symbols on edges represent edge labels. The two sets on the bottom are incoming messages, where the set on the top is the corresponding outgoing message.

then represent the calculation of CTV messages in a compact manner:

$$\mathrm{CTV}^{(l)}_{\mathtt{c}\to\mathtt{v}} = \frac{1}{h_{\mathtt{c},\mathtt{v}}} \sum_{\mathtt{v}'\in\{\mathcal{N}(\mathtt{c})\setminus\mathtt{v}\}} \mathcal{B}^{(l)}_{\mathtt{v}'}, \quad l \geq 1. \tag{12}$$

In words, $\mathrm{CTV}^{(l)}_{\mathtt{c}\to\mathtt{v}}$ is the set of possible values of $\mathtt{v}$ given the incoming VTC messages from the variable nodes in $\{\mathcal{N}(\mathtt{c})\setminus\mathtt{v}\}$, where an example is given in Figure 2a. Note that a CTV message can be of cardinality between 1 and q. We now move to calculate the VTC messages, which are based on the CTV messages. The VTC messages are calculated as follows:

$$\mathrm{VTC}^{(l)}_{\mathtt{v}\to\mathtt{c}} = \mathrm{VTC}^{(0)}_{\mathtt{v}} \bigcap \left(\bigcap_{\mathtt{c}'\in\{\mathcal{N}(\mathtt{v})\setminus\mathtt{c}\}} \mathrm{CTV}^{(l)}_{\mathtt{c}'\to\mathtt{v}} \right), \quad l \geq 1. \tag{13}$$

That is, the VTC message $\mathrm{VTC}^{(l)}_{\mathtt{v}\to\mathtt{c}}$ is the set of symbols containing the *intersection* of the channel information and the incoming CTV messages to variable node $\mathtt{v}$, where an example is given in Figure 2b. A VTC message cardinality can be at most M, as the channel information cardinality is at most M. A decoding failure occurs if at the end of the decoding process there is a VTC message containing more than one symbol.

The decoding process described above reduces to the known iterative decoder proposed for the BEC/QEC [10] when $M = q$. In this case, the passed messages can be either the set containing all q symbols (full erasure) or a set containing the correct symbol only. This greatly simplifies the asymptotic iterative-decoding performance analysis of LDPC codes when used over the BEC [10]. However, apart from erasure/non-erasure messages in the BEC case, there are many other possible message sets in the QPEC decoding process, making the analysis prohibitively complex as q increases. In the following section, we discuss our approach for low-complexity approximate asymptotic decoding performance analysis, which is later shown to capture the exact behaviour quite well.

IV. Density Evolution Analysis

Density evolution analysis of decoding performance is carried out by tracking the asymptotic (in the codeword length) probability of decoding failure at each iteration based on the probabilities of the passed messages [13], [15], [19]. In this section, we use this method for asymptotic performance evaluation of the decoder described in Section III-B. As customary, we assume a randomly constructed Tanner graph with a degree-distribution pair λ and ρ, and a random i.i.d. selection of edge labels distributed uniformly on the non-zero elements of GF(q). In addition, a sufficiently large codeword length is assumed, such that incoming messages to each node at each iteration of the decoding process are statistically independent with high probability (known as the *independence assumption*) [15]. We start with deriving the exact QPEC density-evolution equations, and then move to propose approximate density evolution analysis due to complexity reasons.

Let us denote by $\mathcal{S}_t$, $t = 1, 2, ..., 2^q - 1$, the non-empty subsets of the input alphabet $\mathcal{X}$ (of q symbols), ordered by cardinality and in lexicographical order. These subsets may be passed throughout the decoding process as either VTC or CTV messages (see Section III-B). Denote by $z^{(l)}_t$ the probability that a VTC message at iteration l is $\mathcal{S}_t$. Similarly, denote by $w^{(l)}_t$ the probability that a CTV message at iteration l is $\mathcal{S}_t$. $\mathcal{I}_{\bar{d}}$ (resp. $\mathcal{J}_{\bar{d}}$) will denote an *ordered list* containing $\bar{d} = d - 1$ indices taken (with possible repetitions) from the set of message indices $\{1, 2, ..., 2^q - 1\}$, representing VTC (resp. CTV) messages to a degree-d check (resp. variable) node. Enumerating the edges connected to a check (variable) node 1 to $\bar{d}$, an element in $\mathcal{I}_{\bar{d}}$ (resp. $\mathcal{J}_{\bar{d}}$) is the index of the message on the corresponding edge. For example, there are $15^2 = 225$ ordered lists $\mathcal{I}_2$ for a degree-3 check node when $q = 4$: $(1, 1), (1, 2), (2, 1), (2, 2), ..., (15, 15)$, where the elements of $\mathcal{I}_2$ are the first and second incoming message indices. $\chi_t(\mathcal{I}_{\bar{d}})$ will denote the probability that the VTC messages indexed in $\mathcal{I}_{\bar{d}}$ lead to the CTV message $\mathcal{S}_t$. Similarly, $\eta_t(\mathcal{J}_{\bar{d}})$ will denote the probability that the CTV messages indexed in $\mathcal{J}_{\bar{d}}$ lead to the VTC message $\mathcal{S}_t$. The distributions χ_t and η_t are obtained with respect to the uniform edge labels and the channel information, as demonstrated in the following example.

Example 3: Assume a degree-3 check node and that $q = 4$. Consider $\mathcal{I}_2 = (5, 5)$ and recall that according to our convention, $\mathcal{S}_5 = \{0, \alpha^0\}$. To calculate $\chi_t(\mathcal{I}_2)$, we find all possible outcomes of the sumset $(h_1/h_3) \cdot \{0, \alpha^0\} + (h_2/h_3) \cdot \{0, \alpha^0\}$ where h_1, h_2 and h_3 are i.i.d. random variables uniformly distributed on $\{\alpha^0, \alpha^1, \alpha^2\}$, representing the edge labels. If $h_1 = h_2 = h_3$, the sumset is $\{0, \alpha^0\} + \{0, \alpha^0\} = \{0, \alpha^0\} =$

$\mathcal{S}_5$. On the other hand, if the edge labels are not the same, the sumset is $\left\{0, \alpha^0, \alpha^1, \alpha^2\right\} = \mathcal{S}_{15}$. Therefore, the non-zero χ_t values are $\chi_5 = 1/9$ and $\chi_{15} = 8/9$ in this case. Now consider a degree-3 variable node, where $\mathcal{J}_2 = (6,6)$ and the channel information sets are $\mathcal{S}_5 = \left\{0, \alpha^0\right\}$, $\mathcal{S}_6 = \left\{0, \alpha^1\right\}$ and $\mathcal{S}_7 = \left\{0, \alpha^2\right\}$ (i.e, $M = 2$), each with probability $1/3$. If the channel information is $\mathcal{S}_5$ or $\mathcal{S}_7$, then the intersection between the messages indexed in $\mathcal{J}_2$ and the channel information is $\mathcal{S}_1 = \{0\}$. If the channel information is $\mathcal{S}_6$, the intersection results in $\mathcal{S}_6$. Therefore, we get that for $\mathcal{J}_2 = (2,2)$, the non-zero η_t values are $\eta_1 = 2/3$ and $\eta_6 = 1/3$.

As GF(q) LDPC codes are linear codes, the probability of a given codeword symbol taken from the codebook is $1/q$. This means that a variable node contains a certain set composed of one symbol (i.e., non-erasure) with probability $(1-\varepsilon)/q$. To incorporate this probability in the density-evolution equations, we define the indicator θ_t, which equals 1 if $|\mathcal{S}_t| = 1$ and 0 otherwise. Equipped with the notations above, we get the following compact representation of the QPEC density-evolution equations:

$$w_t^{(l)} = \sum_{i=2}^{d_c} \rho_i \sum_{\mathcal{I}_{i-1}} \left(\prod_{j \in \mathcal{I}_{i-1}} z_j^{(l-1)} \right) \cdot \chi_t\left(\mathcal{I}_{i-1}\right), \tag{14}$$

$$z_t^{(l)} = \varepsilon \sum_{i=2}^{d_v} \lambda_i \sum_{\mathcal{J}_{i-1}} \left(\prod_{j \in \mathcal{J}_{i-1}} w_j^{(l)} \right) \cdot \eta_t\left(\mathcal{J}_{i-1}\right) + \frac{(1-\varepsilon)}{q} \cdot \theta_t. \tag{15}$$

The summation over $\mathcal{I}_{i-1}$ (or $\mathcal{J}_{i-1}$) is understood over all ordered lists containing $i-1$ elements (where i is the node degree) taken from the set of indices $\{1, 2, ..., 2^q - 1\}$. A decoding failure occurs when a variable node is not resolved, i.e., when it contains a set with more than one symbol:

$$p_e^{(l)} = \sum_{t:|\mathcal{S}_t|>1} z_t^{(l)} = 1 - \sum_{t:|\mathcal{S}_t|=1} z_t^{(l)}. \tag{16}$$

Note that for $M = q$, only z_1, w_1, z_{2^q-1} and w_{2^q-1} (i.e., probabilities of full-erasure/non-erasure sets) might be positive. In this case, these probabilities can be represented solely by z_{2^q-1}, as the distributions χ_t and η_t degenerate due to the simple BEC/QEC decoding rules. Equations (14)-(15) can be then readily simplified to obtain the BEC/QEC (one-dimensional) density-evolution equations [10].

Calculating $\chi_t\left(\mathcal{I}_{\bar{d}}\right)$ and $\eta_t\left(\mathcal{J}_{\bar{d}}\right)$ in Equations (14)-(15) might be prohibitive in practice, as the number of subsets increases exponentially with q. To get an estimate of the complexity, consider basic check and variable nodes of degree 3. Given two incoming message sets to the check, calculating the distribution χ_t requires $(q-1)^3$ realizations of edge labels. Because there are $\mathcal{O}\left(2^q\right)$ input-set pairs, we get $\mathcal{O}\left(q^3 \cdot 2^{2q}\right)$ complexity for calculating χ_t. In a similar manner, $\mathcal{O}\left(\binom{q}{M} \cdot 2^{2q}\right)$ operations are required for calculating η_t (the first factor now being the number of possible channel-information sets) for a degree-3 variable node. As an example, about 10^{13} operations are required for the calculation of χ_t when $q = 16$, growing to the order of 10^{23} when $q = 32$. In addition to prohibitive complexity, the exhaustive calculation of χ_t and η_t as demonstrated in Example 3 provides no insights on their behaviour. Moreover, χ_t requires the explicit use of GF(q) arithmetic, making its analysis difficult. These reasons motivate us to propose a more efficient way for estimating the QPEC decoding performance, which we discuss in the following subsection.

A. Cardinality-based approximated density-evolution equations

To overcome the difficulties in evaluating Equations (14)-(15), we propose to track the probability distribution of the VTC/CTV message set *cardinalities*. In our approach, we approximate messages of the same cardinality passed in the decoding process as being equiprobable. The intuition behind this approximation comes from the randomness of the edge labels and the channel output that ”smoothen” most of the non-uniformity that may occur due to the algebraic structure of GF(q). In particular, as the node degrees and the field order grow, the incidence probability of equal-cardinality sets becomes increasingly uniform. The reason is that the entropy of each sum in the sumset performed at check nodes increases with the degree. In addition, the number of sums within the sumset increases with q, increasing the entropy of the sumset result. The approximation was verified empirically as well, where we show in Section V-D that performance analyzed with this assumption gives a very good approximation of the true decoding performance.

To distinguish between message sets and their cardinalities, we use the notation $\mathcal{M}_{\bar{d}}$ to denote an ordered list of $\bar{d} = d-1$ elements taken from $\{1, 2, ..., q\}$, understood as possible incoming message-set cardinalities to a degree d check node. $W_m^{(l)}$ (resp. $Z_m^{(l)}$) will denote the probability that a CTV (VTC) message at iteration l is of cardinality $m = 1, 2, ..., q$. $P_m(\mathcal{M}_{\bar{d}})$ (resp. $Q_m(\mathcal{M}_{\bar{d}})$) will denote the probability that the message-set cardinalities in $\mathcal{M}_{\bar{d}}$ lead to an outgoing CTV (VTC) message of cardinality m. Note that the distributions P_m and Q_m are obtained by summing the probabilities of χ_t and η_t for all t with $|\mathcal{S}_t| = m$, assuming uniform distribution on the input sets with cardinalities in $\mathcal{M}_{\bar{d}}$. Finally, under our approximation, the following equations are derived:

$$W_m^{(l)} \simeq \sum_{i=2}^{d_c} \rho_i \cdot \sum_{\mathcal{M}_{i-1}} \left(\prod_{m' \in \mathcal{M}_{i-1}} Z_{m'}^{(l-1)} \right) \cdot P_m\left(\mathcal{M}_{i-1}\right), \tag{17}$$

$$\begin{aligned} Z_m^{(l)} \simeq & \varepsilon \cdot \sum_{i=2}^{d_v} \lambda_i \cdot \sum_{\mathcal{M}_{i-1}} \left(\prod_{m' \in \mathcal{M}_{i-1}} W_{m'}^{(l)} \right) \cdot Q_m\left(\mathcal{M}_{i-1}\right) \qquad (18) \\ & + (1-\varepsilon) \cdot \delta\left[m-1\right], \end{aligned}$$

where $\delta\left[m\right]$ is the discrete Dirac delta function. The summation over $\mathcal{M}_{i-1}$ is understood over the ordered lists of $i-1$ elements taken from the set of possible incoming message-set cardinalities. This set is $\{1, 2, ..., M\}$ for incoming VTC and $\{1, 2, ..., q\}$ for incoming CTV message-set cardinalities. The initial conditions are $Z_1^{(0)} = 1-\varepsilon, Z_M^{(0)} = \varepsilon$ and $Z_m^{(0)} = 0$ for $m \neq 1, M$. The asymptotic probability of decoding failure at iteration l is the probability of a VTC message-set cardinality larger than 1 at iteration l:

$$p_e^{(l)} = \sum_{m=2}^{q} Z_m^{(l)} = 1 - Z_1^{(l)}. \tag{19}$$

We note here that in our experiments the probability of decoding failure calculated using (19) is virtually the same as (16) even for small q and check-node degree values, such that the cardinality-based equations can be safely used for QPEC performance evaluation. However, though we moved from $\mathcal{O}(2^q)$ possible message sets to q possible message-set cardinalities, we still need efficient ways to calculate P_m and Q_m. A straightforward calculation enumerates χ_t and η_t for $\mathcal{O}(2^{2q})$ realizations of message set pairs, which does not quite solve the complexity problem. Thus we devote the remainder of this section and the next section to efficient calculations, bounding, and approximations for P_m and Q_m. We begin with providing in Section IV-B an exact closed-form expression for Q_m. In Section V we show that finding a closed-form expression for P_m is hard. Therefore, we propose computationally efficient bounds and approximation models for P_m. We later use our models and bounds to determine the QPEC decoding threshold and to design good LDPC codes.

B. Formula for Q_m

$Q_m(\mathcal{M}_{\bar{d}})$ is the probability of an intersection of cardinality m between CTV messages with cardinalities taken from $\mathcal{M}_{\bar{d}}$ and a channel information set of cardinality M, where message sets of the same cardinality are equiprobable. Define $\mathcal{M}_d$ to contain the cardinalities in $\mathcal{M}_{\bar{d}}$ together with the channel information set cardinality M and μ to be the smallest cardinality in $\mathcal{M}_d$, i.e. $\mu \triangleq \min_{m' \in \mathcal{M}_d} m'$. In the following, we find the number of ways to realize the sets in $\mathcal{M}_d$ such that their intersection is of cardinality m, and later take into account the presence of the correct symbol in each set. We begin with the following lemma.

Lemma 2: (Number of ways to get an intersection of cardinality m) Consider d message sets whose cardinalities are in $\mathcal{M}_d$. The number of ways to realize the sets such that their intersection is of cardinality m ($m = 0, 1, ..., \mu$) is:

$$K_m(\mathcal{M}_d; q) = \sum_{s=0}^{\mu-m} (-1)^s \cdot \upsilon_{m+s} \cdot \binom{m+s}{m}, \tag{20}$$

where

$$\upsilon_{m+s} \triangleq \binom{q}{m+s} \cdot \prod_{m' \in \mathcal{M}_d} \binom{q-(m+s)}{m'-(m+s)}. \tag{21}$$

Proof Consider a fixed subset of μ elements taken from a set of q elements. The number of ways to choose d subsets with cardinalities in $\mathcal{M}_d$ such that they all contain the subset of μ elements is $\prod\limits_{m' \in \mathcal{M}_d} \binom{q-\mu}{m'-\mu}$, as we are free to choose only $m' - \mu$ elements for each subset of cardinality m'. Taking into account the number of ways to choose a subset of μ elements, which is $\binom{q}{\mu}$, we have

$$K_\mu = \binom{q}{\mu} \cdot \prod_{m' \in \mathcal{M}_d} \binom{q-\mu}{m'-\mu} = \upsilon_\mu \tag{22}$$

ways to choose the subsets such that their intersection is of cardinality μ. To find K_m for $m = \mu - 1$, we proceed as follows. The number of ways to choose the subsets such that they contain a fixed subset of $\mu - 1$ elements is $\prod\limits_{m' \in \mathcal{M}_d} \binom{q-(\mu-1)}{m'-(\mu-1)}$. However, the subsets may also contain a subset of cardinality μ such that the fixed subset of cardinality $\mu - 1$ is its subset, resulting in overcounting. Since there are $\binom{\mu}{\mu-1} = \mu$ sets of cardinality $\mu - 1$ contained in a set of cardinality μ, we correct for overcounting as follows:

$$\begin{aligned} K_{\mu-1} =& \binom{q}{\mu-1} \cdot \prod_{m' \in \mathcal{M}_d} \binom{q-(\mu-1)}{m'-(\mu-1)} - \mu \cdot \upsilon_\mu \\ =& \upsilon_{\mu-1} - \mu \cdot \upsilon_\mu. \end{aligned} \tag{23}$$

Moving to $\mu - 2$, we first count sets of cardinality $\mu - 2$ with $\upsilon_{\mu-2}$ and then subtract $\binom{\mu-1}{\mu-2} \cdot \upsilon_{\mu-1}$ sets to account for sets of cardinality $\mu - 1$. However, we now over-correct some sets of cardinality μ. We account for that by considering the $\binom{\mu}{\mu-2}$ sets of cardinality $\mu - 2$ contained in a set of cardinality μ to obtain:

$$K_{\mu-2} = \upsilon_{\mu-2} - \binom{\mu-1}{\mu-2} \cdot \upsilon_{\mu-1} + \binom{\mu}{\mu-2} \cdot \upsilon_\mu. \tag{24}$$

Continuing in the same fashion (essentially, we use the inclusion-exclusion principle), we get:

$$K_{\mu-t} = \sum_{i=0}^{t} (-1)^i \cdot \upsilon_{\mu-t+i} \cdot \binom{\mu-t+i}{\mu-t}, \tag{25}$$

for $t = 0, 1, ..., \mu$. Index shifting leads to the desired result. ∎

We are now ready to provide a formula for Q_m. Lets us denote by $\mathcal{M}_d - 1$ the ordered list obtained by subtracting 1 from each number (set cardinality) in $\mathcal{M}_d$.

Theorem 3: (Formula for Q_m)

$$Q_m\left(\mathcal{M}_{\bar{d}}\right) = \begin{cases} \dfrac{K_{m-1}(\mathcal{M}_d - 1; q-1)}{\prod\limits_{m' \in \mathcal{M}_d} \binom{q-1}{m'-1}}, & \text{if } \mu > 1 \\ \delta\left[m-1\right], & \text{otherwise.} \end{cases} \tag{26}$$

Proof We use K_{m-1}, $\mathcal{M}_d - 1$ and $q - 1$ as we can choose effectively $m' - 1$ elements for each subset of cardinality m', as the correct symbol appears in the subsets. We then normalize by the number of subsets with cardinalities $m' - 1$ taken from a set of $q - 1$ elements to obtain a probability distribution. Note that when $\mu = 1$ the intersection is necessarily of cardinality 1, such that that $Q_1 = 1$. ∎

V. Bounds and approximations for P_m

$P_m(\mathcal{M}_{\bar{d}})$ in Equation (17) is the probability that the sumset of the sets with cardinalities in $\mathcal{M}_{\bar{d}}$ is of cardinality m, where sets of the same cardinality are equiprobable and the edge labels are uniformly distributed. Considering all possible realizations of the messages becomes intractable as the field size or the node degree increase. The major reason for the difficulty in calculating P_m (unlike Q_m) is that it involves GF(q) arithmetic. Thus, finding a closed-form expression for P_m is hard, see e.g. the discussion on sumsets in [18], [20], [21]. Because of that, we seek instead efficient bounds and approximations for P_m. Let $\mathcal{I}_{\bar{d}}$ contain indices of arbitrary message sets whose cardinalities are in $\mathcal{M}_{\bar{d}}$. Denote $\kappa \triangleq \max_{m' \in \mathcal{M}_{\bar{d}}} m'$ as the maximal number (set cardinality) in $\mathcal{M}_{\bar{d}}$. In addition, denote $N \triangleq \prod_{m' \in \mathcal{M}_{\bar{d}}} m'$ as the number of sums in the calculation of $\sum_{j \in \mathcal{I}_{\bar{d}}} \mathcal{S}_j$.

Example 4: Assume that $q = 4$ and $\bar{d} = 2$. If $\mathcal{M}_{\bar{d}} = \{2, 3\}$, then the first element in $\mathcal{I}_{\bar{d}}$ can be between 5 and 10, and the second element can be between 11 and 14.

A. Upper and lower bounds on P_m using additive combinatorics

In this subsection we derive bounds on the cardinality of the sumset $\sum_{j \in \mathcal{I}_{\bar{d}}} \mathcal{S}_j$. These bounds will be a function of the message-set cardinalities $\mathcal{M}_{\bar{d}}$, such that they are universal for all realizations of sets adhering to the cardinalities in $\mathcal{M}_{\bar{d}}$. We begin with simple lower and upper bounds.

Lemma 4: (Simple bounds on a sumset cardinality [18])

$$\kappa \leq \left| \sum_{j \in \mathcal{I}_{\bar{d}}} \mathcal{S}_j \right| \leq \min(q, N). \tag{27}$$

The following lemma provides a sufficient condition for attaining the maximal sumset cardinality q.

Lemma 5: (Sufficient condition for the sumset of cardinality q [18]) If there are $m, m' \in \mathcal{M}_{\bar{d}}$ (where m and m' are taken from two different positions in $\mathcal{M}_{\bar{d}}$) such that $m + m' > q$, then $\left| \sum_{j \in \mathcal{I}_{\bar{d}}} \mathcal{S}_j \right| = q$.

For later use, we say that the *q-condition* holds if the condition of Lemma 5 is satisfied. Note that this condition can be satisfied only if $M > q/2$. We now proceed to obtain improved lower bounds on the sumset cardinality, using the following two theorems.

Theorem 6: (Cauchy-Davenport Theorem [18]) Consider the finite field GF(p), p prime. Let $\mathcal{S}_a$ and $\mathcal{S}_b$ be two non-empty subsets of GF(p). Then:

$$|\mathcal{S}_a + \mathcal{S}_b| \geq \min(p, |\mathcal{S}_a| + |\mathcal{S}_b| - 1). \tag{28}$$

The following theorem by Károlyi provides an extension of the Cauchy-Davenport theorem to finite groups.

Theorem 7: (Károlyi's theorem for finite groups [22]) Let $\mathcal{S}_a$ and $\mathcal{S}_b$ be two non-empty subsets of a finite group G. Denote by $p(G)$ the smallest prime factor of $|G|$. Then:

$$|\mathcal{S}_a + \mathcal{S}_b| \geq \min(p(G), |\mathcal{S}_a| + |\mathcal{S}_b| - 1). \tag{29}$$

This theorem can be used for extending the inequality (28) to extension fields, as we have in the following theorem.

Theorem 8: (Improved sumset cardinality bounds) Denote by p the prime factor of q. Then:

$$\max\left(\kappa, \min\left(p, \sum_{m' \in \mathcal{M}_{\bar{d}}} m' - \bar{d} + 1\right)\right) \leq \left| \sum_{j \in \mathcal{I}_{\bar{d}}} \mathcal{S}_j \right| \tag{30}$$
$$\leq \min(q, N).$$

Proof This theorem is proved by Lemma 4 and Theorem 7, followed by induction on the number of subsets (see e.g. [23] for the proof technique when q is prime). ▮

The bounds of Theorem 8 are sharp (i.e., there exist subsets $\mathcal{S}_j$ with cardinalities in $\mathcal{M}_{\bar{d}}$ such that the bounds are attained) [18]. We will denote by B_L and B_U the lower and upper bounds of inequality (30), respectively. We use these bounds to derive two bounding distributions $P_m^{(\max)}$ and $P_m^{(\min)}$: the former to bound the output set cardinalities from above, and the latter from below. To get $P_m^{(\max)}$, the sumset is assumed as of cardinality B_U with probability 1, unless the q-condition is satisfied.

$$P_m^{(\max)} = \begin{cases} \delta[m - q], & \text{if the } q\text{-condition holds} \\ \delta[m - B_U], & \text{otherwise.} \end{cases} \tag{31}$$

In a similar manner, $P_m^{(\min)}$ is calculated using the lower bound B_L on the sumset cardinality:

$$P_m^{(\min)} = \begin{cases} \delta[m - q], & \text{if the } q\text{-condition holds} \\ \delta[m - B_L], & \text{otherwise.} \end{cases} \tag{32}$$

The importance of $P_m^{(\max)}$ resp. $P_m^{(\min)}$ is that using them in the density evolution iteration in place of the true P_m gives a lower resp. upper bound on the asymptotic probability of decoding failure (19) calculated using the cardinality-based density-evolution equations.

Going beyond the bounds above to a potentially tighter characterization of P_m, in the remainder of the section we propose two low-complexity approximation models for P_m. We begin with a simple balls-and-bins model, and later refine it with a tighter model. Finally, we compare the bounds above with the proposed approximation models.

B. The balls-and-bins model

The major difficulty in calculating P_m exactly is its dependence on the structure of the finite-field arithmetic. Going around this difficulty, we propose a pure-probabilistic approximation of P_m using the *balls-and-bins model* [24]. In this model, balls are placed independently and uniformly at random to bins, where we are usually interested in the distribution of the number of non-empty bins once all the balls were placed. Motivated by the randomness induced by the random edge labels, we propose to consider the N sums in the calculation of the sumset as the balls, and the q elements of GF(q) as the bins. This way, P_m is modeled as the probability of m non-empty bins after the N balls were placed. As a consequence, the use of GF(q) arithmetic is not required when the balls-and-bins model is used.

The balls-and-bins model is an *absorbing* Markov process with $q + 1$ possible *states*, with state m ($m = 0, 1, ..., q$) corresponding to m non-empty bins out of q. The absorbing state is q, as once q bins are non-empty the number of non-empty bins cannot change. The $(q + 1) \times (q + 1)$ Markov matrix describing this process takes a simple form, since we can either stay at state m or move to state $m + 1$. Denoting the Markov matrix as $\mathbf{\Gamma}_{\text{balls}}$, its entries are:

$$(\mathbf{\Gamma}_{\text{balls}})_{m,m} = \frac{m}{q}, (\mathbf{\Gamma}_{\text{balls}})_{m,m+1} = 1 - \frac{m}{q}, \tag{33}$$

where the remaining entries are zeros. That is, if the current state is m, then a ball is placed in a one of the m non-empty bins with probability m/q, and is placed in a different bin with probability $1 - m/q$. Let us denote by $\boldsymbol{g}^{(N)} = \left(g_0^{(N)}, g_1^{(N)}, ..., g_q^{(N)}\right)$ the probability distribution on the states defined by $\boldsymbol{\Gamma}_{\text{balls}}$, where $g_m^{(N)}$ is the probability of state m after the N balls were placed. According to the Markov property, $\boldsymbol{g}^{(N)} = \boldsymbol{g}^{(0)} \cdot \boldsymbol{\Gamma}_{\text{balls}}^N$ (where $\boldsymbol{\Gamma}_{\text{balls}}^N$ is $\boldsymbol{\Gamma}_{\text{balls}}$ raised to power N). As $\boldsymbol{g}^{(0)} = (1, 0, ..., 0)$ (i.e., the bins are empty at the beginning), $\boldsymbol{g}^{(N)}$ is simply the first row of $\boldsymbol{\Gamma}_{\text{balls}}^N$. Finally, using the q-condition (Lemma 5) and the lower bound B_L (see Section V-A), we define the following approximation model for P_m:

$$P_m^{(\text{balls})} = \begin{cases} 0, & \text{if } m < B_L \\ \delta\left[m-q\right], & \text{if the } q\text{-condition holds} \\ \dfrac{g_m^{(N)}}{\sum\limits_{m'=B_L}^{q} g_{m'}^{(N)}}, & \text{otherwise.} \end{cases} \tag{34}$$

The expected number of balls required to get into the absorbing state q (when starting at state 0) is $q \ln q + q\gamma$ (up to $\mathcal{O}\left(1/\left(2q\right)\right)$ terms), where $\gamma \approx 0.577$ is the *Euler-Mascheroni constant* [24]. That is, all the bins will be non-empty on average when $N \approx q \ln q + q \cdot \gamma$, which can be thought as the probabilistic extension of the q-condition to the balls-and-bins model. For such N values, the sumset cardinality approximated using the balls-and-bins model is expected to be q and $\boldsymbol{g}^{(N)}$ degenerates to the absorbing distribution $(0, \ldots, 0, 1)$. Therefore, $\boldsymbol{\Gamma}_{\text{balls}}^N$ should be calculated in practice for values of N up to approximately $q \ln q + q\gamma$, even for high-degree check nodes.

C. The union model

In the previous sub-section, we modeled P_m using the balls-and-bins model, where each ball (which corresponds to an element obtained by a sum within the sumset) is independent of the other balls. In this part, we improve this approximation by exploiting an important property of the N sums within the sumset: they can be divided into N/κ sets of κ (distinct) elements. This is proved by viewing the sums as generated by one element from the maximal-cardinality subset (of cardinality κ) and elements from the remaining subsets. This observation leads us to suggest a refined version of the balls-and-bins model, which we term as the *union model*. In this model, the probability of a sumset of cardinality m is modeled as the probability that the union of N/κ random sets with cardinality κ each results in a set of cardinality m. In view as balls-and-bins, it is the probability of m non-empty bins after κ groups of N/κ balls are placed into the q bins, where the balls in each group are placed uniformly at random into κ *distinct* bins.

Let us denote by $\boldsymbol{u}^{(N/\kappa)} = \left(u_0^{(N/\kappa)}, u_1^{(N/\kappa)}, ..., u_q^{(N/\kappa)}\right)$ the probability distribution on the $q+1$ states after N/κ groups of balls were placed into the bins. That is, $u_m^{(N/\kappa)}$ is the probability of state m after N/κ groups of κ balls each were placed in the bins according to the union model. The transition probability $P_{\text{union}}\left(m \to m'\right)$ from state m to state m' is equivalent to the probability that the union of a random set of cardinality m with a random set of cardinality κ is of cardinality m' (given that the set elements are taken from q elements). To calculate this probability, denote by $\mathcal{A}$ a set of cardinality m and by $\mathcal{B}$ a set of cardinality κ. We have:

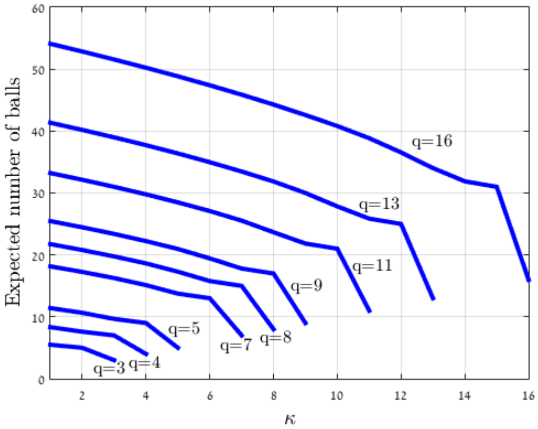


Fig. 3: The expected number of balls required for getting q non-empty bins in the union model.

$$\begin{aligned} P_{\text{union}}\left(m \to m'\right) &= \Pr\left(\left|\mathcal{A} \cup \mathcal{B}\right| = m'\right) \\ &= \Pr\left(\left|\mathcal{A} \cap \mathcal{B}\right| = m + \kappa - m'\right), \end{aligned} \tag{35}$$

where we used the inclusion-exclusion principle. Thus, we can equivalently find the probability that the *intersection* of the sets $\mathcal{A}$ and $\mathcal{B}$ is of cardinality $m + \kappa - m'$. Recall that $K_{m+\kappa-m'}\left(m, \kappa\right)$ (see Lemma 2, q is omitted for brevity) is the number of ways to obtain such an intersection cardinality. Dividing $K_{m+\kappa-m'}\left(m, \kappa\right)$ by the number of possible realizations of elements in the sets provides the desired probability $P_{\text{union}}\left(m \to m'\right)$. Therefore, the entries of the Markov matrix associated with the union model are:

$$\left(\boldsymbol{\Gamma}_{\text{union}}\right)_{m,m'} = \frac{K_{m+\kappa-m'}\left(m, \kappa\right)}{\binom{q}{m} \cdot \binom{q}{\kappa}}. \tag{36}$$

It is not hard to check that for $\kappa = 1$, $\boldsymbol{\Gamma}_{\text{union}}$ reduces to $\boldsymbol{\Gamma}_{\text{balls}}$ (defined in (33)). As before, the Markov property implies that $\boldsymbol{u}^{(N/\kappa)}$ is simply the first row of $\boldsymbol{\Gamma}_{\text{union}}^{N/\kappa}$. Finally, the following approximation for P_m is based on the union model:

$$P_m^{(\text{union})} = \begin{cases} 0, & \text{if } m < B_L \\ \delta\left[m-q\right], & \text{if the } q\text{-condition holds} \\ \dfrac{u_m^{(N/\kappa)}}{\sum\limits_{m'=B_L}^{q} u_{m'}^{(N/\kappa)}}, & \text{otherwise} \end{cases} \tag{37}$$

To obtain the expected number of balls required to get into the absorbing state q, we use the *fundamental matrix* [25] associated with an absorbing Markov chain. In our case, this matrix is $\boldsymbol{\Phi}_{\text{union}} = \left(\mathbf{I}_q - \mathbf{Q}_{\text{union}}\right)^{-1}$ where $\mathbf{I}_q$ is the identity matrix of dimensions $q \times q$ and $\mathbf{Q}_{\text{union}}$ is the upper-left $q \times q$ sub-matrix of $\boldsymbol{\Gamma}_{\text{union}}$. The expected number of groups of balls required to get into the absorbing state q when starting with state i_0 is the i_0^{th} entry of the vector $\boldsymbol{\Phi}_{\text{union}}\mathbf{1}$, where $\mathbf{1}$ is a column vector whose entries are all 1 [25] (to get the expected number of *balls*, we multiply by κ). In Figure 3, the expected number of balls required for getting from state 0 to state q is given as a function of q and κ. As mentioned earlier (see Section V-B), this expected number can be used for extending the q-condition to the union model. Note that for $\kappa = 1$ the union model is essentially the balls-and-bins model and we have $\mathbf{\Phi}_{\text{union}}\mathbf{1}\,(i=0) \approx q \ln q + q\gamma$ as we saw earlier.

D. Comparison of the bounds and approximations

In this part, we verify the tightness of our approximations by comparing the *decoding threshold* [15] obtained from the exact and the approximate (cardinality-based) density-evolution equations. The QPEC decoding threshold (for a given degree-distribution pair), denoted ε_{th}, is the maximal partial-erasure probability ε such that the probability of decoding failure tends to zero. Its operational meaning is the robustness of the iterative decoder to partially-erased codeword symbols, i.e., the fraction of partially-erased codeword symbols that the decoder can tolerate. In Figure 4, we plot the exact and approximate decoding threshold values (using the bounds and models for P_m) for several values of q and M for the regular $(3, 6)$ LDPC code ensemble (of rate $1/2$). We note that the exact threshold for $q = 16$ is not provided in Figure 4 due to complexity reasons. When $M = q$, the QPEC density-evolution equations are equivalent to the BEC/QEC density-evolution equation (see Section IV). In this case, all the models and bounds give the exact threshold, which is 0.429.

According to Figure 4, the upper bound on the threshold calculated using the cardinality-based equations becomes loose as q increases. This is due to the dependency of the sumset cardinality lower bound (see Theorem 8) on the smallest prime factor of q, which is 2 for binary fields. This makes the threshold upper bound for such fields less tight compared to prime fields. The lower bound is also somewhat loose, as it corresponds to the upper bound on the sumset cardinality that depends on the number of sums in the sumset. However, the bounds on the sumset cardinality are sharp (see Section V-A), so it is difficult to improve the bounds shown in Figure 4. On the other hand, the balls-and-bins model and the union model provide good approximations of the exact threshold, and they are significantly tighter than the bounds. Recall that these approximations can be calculated efficiently, making the models especially attractive for large values of q. We deduce from Figure 4 an interesting result: not considering the algebraic structure of the field when using the approximation models leads on average to *smaller* sumset cardinalities compared to the exact calculation of the susmset. If, as we conjecture, the approximation models give indeed upper bounds on the threshold, the uncertainty interval of the exact threshold is relatively small, and it becomes smaller as M approaches q.

Figure 4 suggests a potential application of the QPEC to speed up the read process in measurement channels. As an example, suppose that $q = 8$ and $M = 4$. The decoding threshold in this case is approximately 0.59. Thus, instead of performing $q-1 = 7$ comparative measurements to completely read the stored symbol, in 59% of the cells we may perform only one measurement (yielding $q/2 = 4 = M$ uncertainty). In terms of read rate, we now need only 3.46 measurements on average, improving the read rate by more than 50%.

VI. Code Design using Linear Programming

The design of good LDPC codes for the QPEC using the exact density-evolution equations (14)-(15) is difficult due to their $\mathcal{O}(2^{2q})$ dimensionality (see Section IV). Motivated by the efficiency and the good approximations obtained using the cardinality-based approach, we propose two methods for linear programming (LP) optimization of degree distributions. In Section VI-A, we present a threshold-oriented iterative optimization process. In Section VI-B, we use the union model to achieve a target of small decoding-failure probability.

A. Iterative QPEC code design

Let us denote by $\varepsilon^{\text{c}}_{\text{th}}$ the approximate decoding threshold obtained using the cardinality-based equations (17)-(18). In this subsection, we propose two LP optimization methods for obtaining a degree-distribution pair with a desired $\varepsilon^{\text{c}}_{\text{th}}$ value. Recall that $W_q^{(l)}$ denotes the probability of a CTV message of cardinality q at iteration l of the decoding process when the cardinality-based equations are used (see Section IV-A). Assuming a QPEC with $M > q/2$, a sufficient condition for an outgoing CTV message to be of cardinality q is the q-condition, meaning that there is at least one pair of incoming VTC messages whose sum of cardinalities exceeds q (see Lemma 5). Therefore, $W_q^{(l)}$ is bounded from below by the probability that at least two incoming VTC messages are of cardinality M, since $2M > q$ when $M > q/2$. As $Z_M^{(l-1)}$ denotes the probability for a VTC message of cardinality M at iteration $l-1$, we get:

$$\begin{aligned} W_q^{(l)} &\geq \sum_{i=2}^{d_c} \rho_i \sum_{j=2}^{i-1} \binom{i-1}{j} \left(Z_M^{(l-1)}\right)^j \left(1 - Z_M^{(l-1)}\right)^{i-1-j} \\ &= 1 - \rho\left(1 - Z_M^{(l-1)}\right) - Z_M^{(l-1)} \rho'\left(1 - Z_M^{(l-1)}\right), \end{aligned} \tag{38}$$

where $\rho'(x)$ denotes the derivative of the polynomial $\rho(x)$ with respect to x. A sufficient condition for obtaining VTC messages of cardinality M is that a variable node is a partial erasure and all its incoming CTV messages are of cardinality q. Therefore,

$$Z_M^{(l-1)} \geq \varepsilon \sum_{i=1}^{d_v} \lambda_i \left(W_q^{(l-1)}\right)^{i-1} = \varepsilon \lambda\left(W_q^{(l-1)}\right). \tag{39}$$

$\lambda(x)$ is an increasing function of x for $x \geq 0$, since $\lambda(x)$ is a polynomial with non-negative coefficients. Note that both $W_q^{(l)}$ and the right-hand side of (38) are non-negative as probabilities. Thus, according to (38),

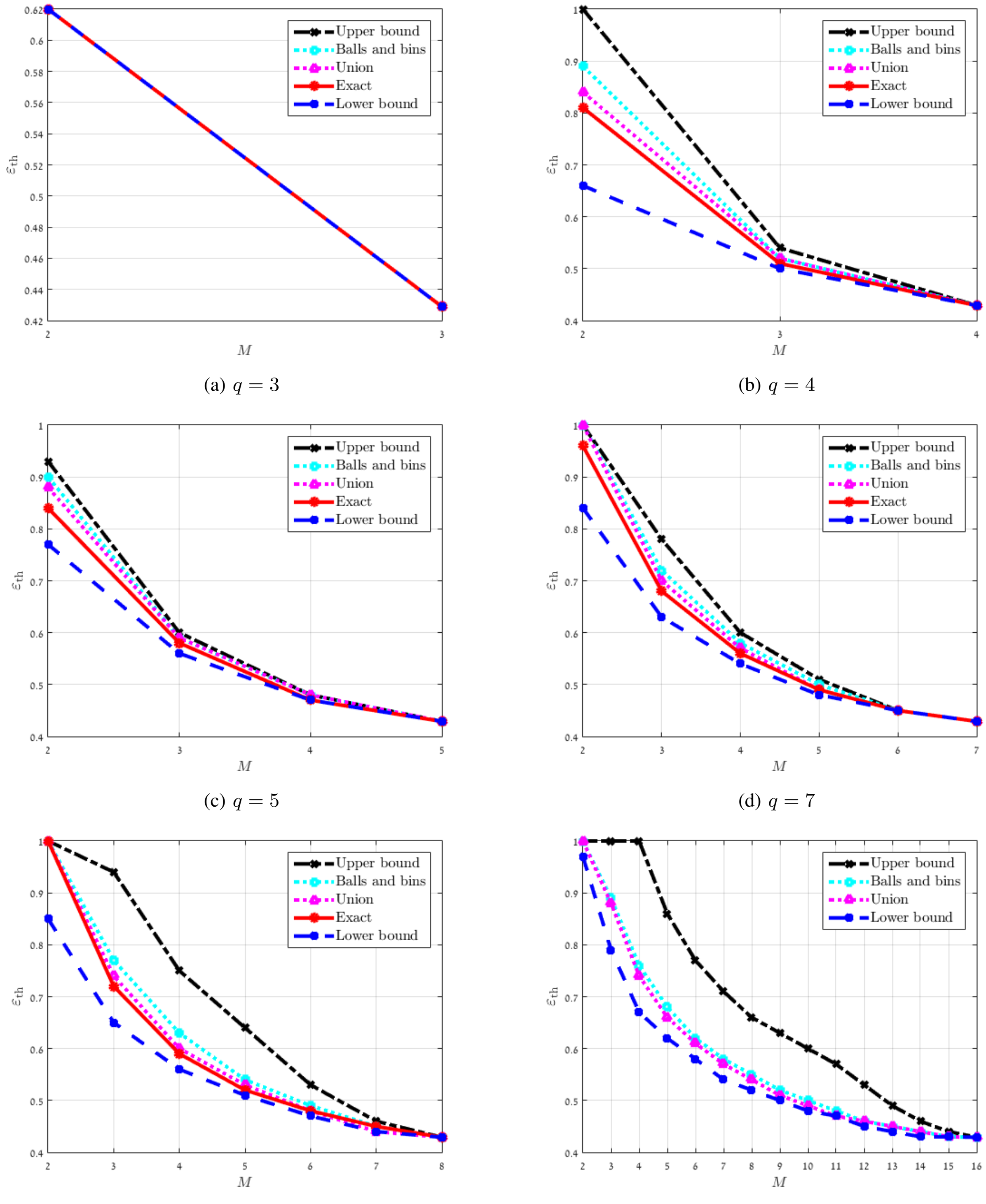
$$\lambda\left(W_q^{(l-1)}\right) \geq \lambda\left(1 - \rho\left(1 - Z_M^{(l-2)}\right) - Z_M^{(l-2)} \rho'\left(1 - Z_M^{(l-2)}\right)\right). \tag{40}$$

Finally, by combining (39) and (40), we get:

$$Z_M^{(l)} \geq \varepsilon \lambda\left(1 - \rho\left(1 - Z_M^{(l-1)}\right) - Z_M^{(l-1)} \rho'\left(1 - Z_M^{(l-1)}\right)\right), \tag{41}$$

with the initial condition $Z_M^{(0)} = \varepsilon$.

The inequality in (41) applies to any $M > q/2$, for all q (which is prime or prime power), and depends solely on the



(a) $q=3$

(b) $q=4$

(c) $q=5$

(d) $q=7$

(e) $q=8$

(f) $q=16$

Fig. 4: The QPEC threshold of the regular $(3,6)$ LDPC code ensemble, as a function of q and M.

degree distributions $\lambda(x)$ and $\rho(x)$. Recall that in the case of the BEC, we have an equality rather than an inequality, and without the additional term $-Z_M^{(l-1)} \cdot \rho' \left(1 - Z_M^{(l-1)}\right)$. This term leads to an upper bound on $\varepsilon_{\text{th}}^{\text{c}}$, as we will see. Define the function $h_\varepsilon(x) = \varepsilon\lambda\left(1 - \rho\left(1-x\right) - x\rho'\left(1-x\right)\right)$ (which is the right-hand side of the inequality in (41) with $Z_M^{(l-1)}$ replaced by x), and denote by $h_\varepsilon^l(x)$ the l^{th} composition of $h_\varepsilon(x)$ with itself. We begin with the following lemma.

Lemma 9:

1) $Z_M^{(l)} \geq h_\varepsilon^l\left(\varepsilon\right), \qquad l \geq 1.$
2) $\lim\limits_{l\to\infty} h_\varepsilon^l\left(\varepsilon\right)$ exists and is an increasing function of ε.

The proof of this lemma is provided in Appendix B. Observing that $h_{\varepsilon}^{l}(\varepsilon)=0$ for $\varepsilon=0$ and using the second part of Lemma 9, we are able to define the following value:

$$\varepsilon^{*}=\sup\left\{\varepsilon\in[0,1]:\lim_{l\to\infty}h_{\varepsilon}^{l}(\varepsilon)=0\right\}. \tag{42}$$

Note that ε^{*} is defined with respect to a certain degree-distribution pair λ and ρ. This definition of ε^{*} leads to an upper bound on $\varepsilon_{\text{th}}^{\text{c}}$.

Theorem 10: For a QPEC with $M>q/2$, $\varepsilon_{\text{th}}^{\text{c}}\le\varepsilon^{*}$.

Proof $Z_{M}^{(l)}$ is bounded from below by a strictly positive value for all l when $\varepsilon>\varepsilon^{*}$, according to Lemma 9 and the definition of ε^{*} in (42). Since the probability of decoding failure according to the cardinality-based approach (19) in this case is necessarily non-zero, $\varepsilon_{\text{th}}^{\text{c}}$ cannot exceed ε^{*}. ∎

For the formulation of an LP optimization, we derive an equivalent definition for ε^{*}, by extending the fixed-point characterization of the BEC threshold [19].

Theorem 11: For a QPEC with $M>q/2$,

$$\varepsilon^{*}=\sup\left\{\varepsilon\in[0,1]:x=h_{\varepsilon}(x)\ \text{ has no solution } x \text{ in } (0,1]\right\}. \tag{43}$$

The proof of this theorem is similar to the proof of Theorem 3.59 in [10], and is omitted. We now formulate an LP optimization for determining good (in terms of code rate) variable-node degree distribution $\lambda(x)$ for given $\rho(x)$ and ε^{*} assuming that $M>q/2$. A maximum constraint d_v on variable-node degrees is set, as usual [10], to control implementation complexity and convergence speed. According to the ε^{*} equivalent definition (43), the condition for degree distributions whose threshold is upper bounded by ε^{*} is that $h_{\varepsilon^{*}}(x)-x\le 0$ for $x\in(0,1]$. This leads us to formulate an LP optimization for the QPEC, where maximal rate is sought under the constraint that $\varepsilon_{\text{th}}^{\text{c}}$ is upper bounded by ε^{*}:

$$\max_{\lambda}\left\{\sum_{i=2}^{d_v}\frac{\lambda_i}{i}:\lambda_i\ge 0,\sum_{i=2}^{d_v}\lambda_i=1,h_{\varepsilon^{*}}(x)-x\le 0,x\in(0,1]\right\}. \tag{44}$$

We term the LP optimization in (44) as QPEC* LP. Note that the decoding threshold increases as M decreases, such that the degree distributions obtained by QPEC* LP provide at least the same threshold for a QPEC with $M\le q/2$. The difference between the known BEC (or QEC) LP [10] and QPEC* LP is in using in (44) the function $h_{\varepsilon^{*}}(x)$ specially developed for the QPEC, instead of the function $f_{\varepsilon}(x)=\varepsilon\cdot\lambda\left(1-\rho\left(1-x\right)\right)$ derived from the BEC density-evolution equation.

The QPEC* LP optimization provides a degree-distribution pair with $\varepsilon_{\text{th}}^{\text{c}}$ upper-bounded by ε^{*}. This suggests the following strategy for obtaining degree distributions with a desired value of $\varepsilon_{\text{th}}^{\text{c}}$. Choose ε^{*} that is larger than the desired $\varepsilon_{\text{th}}^{\text{c}}$, and solve the QPEC* LP optimization. Find $\varepsilon_{\text{th}}^{\text{c}}$ of the optimized degree distributions using the cardinality-based density-evolution equations (where the union model is suggested for large q). If the threshold is smaller than $\varepsilon_{\text{th}}^{\text{c}}$, increase ε^{*} and repeat the process. Otherwise, decrease ε^{*} and repeat the process. An alternative design method using previously known theoretical tools is to seek the desired $\varepsilon_{\text{th}}^{\text{c}}$ using the BEC LP optimization. Because the BEC is a degraded version of the QPEC, here we will choose a target BEC threshold $\varepsilon_{\text{th}}^{\text{BEC}}$ smaller than the desired $\varepsilon_{\text{th}}^{\text{c}}$. We then similarly calculate $\varepsilon_{\text{th}}^{\text{c}}$ of the resulting degree distributions, and decrease/increase the BEC threshold as needed (note that the BEC LP approach is valid for $M\le q/2$ as well).

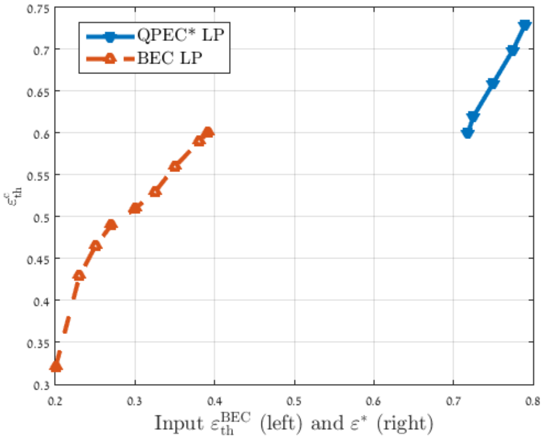


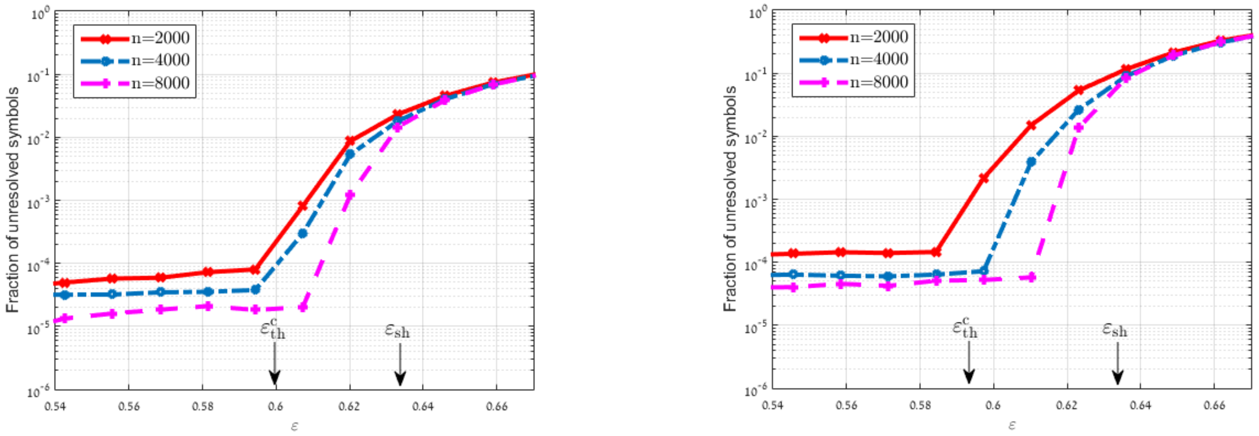
Fig. 5: An example of the iterative optimization process ($q=3,M=2$).

It turns out that using the QPEC* LP approach can result in better codes compared to the BEC optimization. In the sequel we show this by numerical examples. The intuition behind this improvement is that the QPEC* LP optimization better captures the decoding performance for QPECs with $M<q$. We now show the benefit of the new QPEC LP optimization in achieving better code ensembles than those obtained using the BEC LP optimization. As an example, assume that $\rho(x)=x^{5}$, $d_v=5$ and the desired $\varepsilon_{\text{th}}^{\text{c}}$ is 0.6. We concentrate here on QPECs with $M=\lfloor q/2\rfloor+1$ for several values of q (this value of M is the smallest satisfying $M>q/2$). An illustration of the iterative optimization process is provided in Figure 5. The plot shows the sequence of optimization runs of the QPEC* optimizer (right), and the sequence of runs for the BEC optimizer (left). The QPEC* LP approaches the target of $\varepsilon_{\text{th}}^{\text{c}}=0.6$ from above, and the BEC LP from below. Note the approximate linear behaviour of $\varepsilon_{\text{th}}^{\text{c}}$ as a function of ε^{*}, rendering the iterated QPEC* LP as a simpler way for code design. As a consequence, reaching the desired QPEC threshold took typically fewer optimization instances with the QPEC* optimizer than with the BEC optimizer.

The optimized variable degree distributions and their corresponding rates are listed in Table I, together with the values of ε^{*} and $\varepsilon_{\text{th}}^{\text{BEC}}$ resulting in $\varepsilon_{\text{th}}^{\text{c}}$. When comparing the results, we observe that for all the parameters checked the rates achieved by the QPEC* optimizer are strictly better than the rates resulting from the BEC optimizer. Another interesting observation is that in some cases the BEC optimizer required a $\lambda(x)$ polynomial with more non-zero coefficients than the QPEC optimizer. To evaluate the decoding performance in the practical setting of finite-length codes, we constructed random parity-check matrices for varying code lengths and performed 80 iterative decoding iterations using the message-passing de-

TABLE I: Iterative optimization results for $\rho(x) = x^5$, $d_v = 5$ ($\varepsilon^{\mathrm{c}}_{\mathrm{th}} = 0.6$).

q	M	QPEC* LP $\lambda(x)$	Rate	ε^*	BEC LP $\lambda(x)$	Rate	$\varepsilon^{\mathrm{BEC}}_{\mathrm{th}}$
3	2	$0.644x + 0.356x^4$	0.576	0.718	$0.517x + 0.099x^2 + 0.384x^3$	0.569	0.391
4	3	$0.193x + 0.807x^4$	0.354	0.778	$0.157x + 0.843x^4$	0.325	0.532
5	3	$0.489x + 0.511x^4$	0.519	0.751	$0.437x + 0.056x^2 + 0.507x^4$	0.508	0.464
7	4	$0.372x + 0.628x^4$	0.465	0.763	$0.345x + 0.655x^4$	0.451	0.492
8	5	$0.46x + 0.54x^4$	0.507	0.749	$0.413x + 0.587x^4$	0.485	0.48
16	9	$0.422x + 0.578x^4$	0.489	0.754	$0.385x + 0.615x^4$	0.471	0.487



(a) QPEC* LP.

(b) BEC LP.

Fig. 6: Finite-length decoding performance of the LP optimized degree distributions designed for $q = 8, M = 5$ in Table I (rate 0.507). $\varepsilon_{\mathrm{sh}} = 0.637$ is the Shannon limit for these QPEC parameters with rate 0.507.

coder described in Section III-B. In Figure 6, we compare the performance of the QPEC* LP and BEC LP optimized degree distributions for $q = 8$ and $M = 5$. For a fair comparison, we set $\varepsilon^{\mathrm{BEC}}_{\mathrm{th}}$ to 0.472 to obtain a degree-distribution pair with the same rate (0.507) as in the QPEC* LP optimization, leading to $\varepsilon^{\mathrm{c}}_{\mathrm{th}} = 0.594$ in the BEC LP optimization. The QPEC* optimized degree-distribution pair exhibits notably superior decoding performance, for complexity similar to the BEC LP. That is, the QPEC* LP, tailored for the QPEC, better captures the channel behaviour compared to the use of the BEC LP.

B. Code design using the union model

In this part, we use the union model to obtain degree-distribution pairs that achieve a small probability of decoding failure. We begin with a given degree-distribution pair $(\lambda(x), \rho(x))$ and a desired (small) probability of decoding failure p_{tar}, such that there exists an iteration number L satisfying $p_e^{(L)} \le p_{\mathrm{tar}} < p_e^{(L-1)}$. Our aim is to find $\tilde{\lambda}(x)$ that either achieves a lower target of decoding-failure probability, or achieves the same probability in fewer decoding iterations. To find such a variable degree distribution, we adapt the optimization method suggested in [19] to the QPEC, as follows. Define $A_{l,i}$ as the probability of decoding failure at iteration l assuming that we use $\lambda(x)$ at the first $l-1$ iterations followed by the use of the node variable degree distributions with its mass on the degree i at iteration l. Note that $p_e^{(l)}$ is obtained as the following sum

$$p_e^{(l)} = \sum_{i=2}^{d_v} A_{l,i} \cdot \lambda_i. \tag{45}$$

We also define the probability of decoding failure due to degree-i variable nodes assuming that $\lambda(x)$ is used for the first $l-1$ iterations and that $\tilde{\lambda}(x)$ is used at iteration l as $\tilde{p}_e^{(l)}$, which is the right-hand size of (45) with λ_i replaced with $\tilde{\lambda}_i$ The number of decoding iterations required to obtain the probability of decoding failure $p_e^{(L)}$ is approximated as [19]

$$G\left(\tilde{\lambda}\right) \simeq \sum_{l=1}^{L} \frac{\tilde{p}_e^{(l)} - p_e^{(l)}}{p_e^{(l-1)} - p_e^{(l)}}, \tag{46}$$

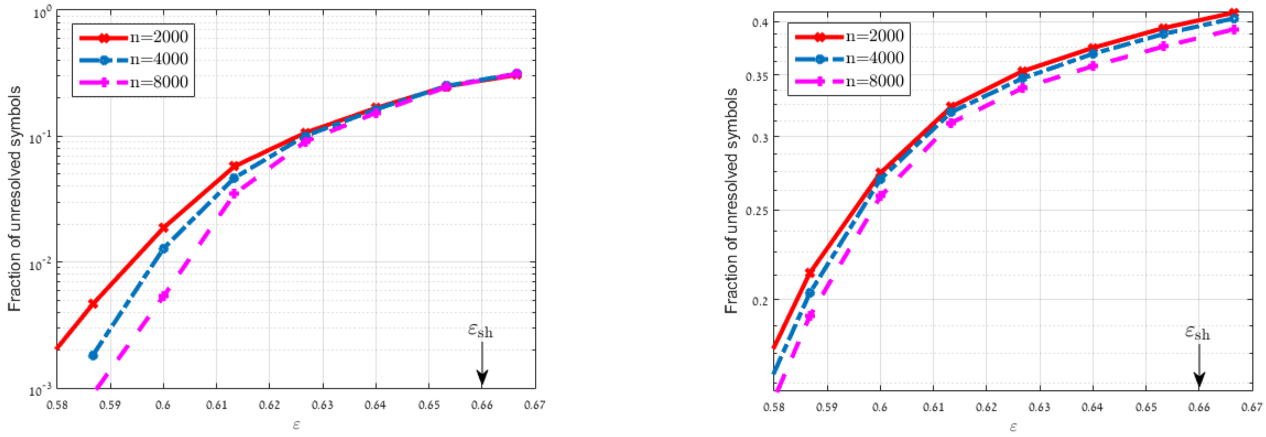
assuming that $\tilde{\lambda}(x)$ and $\lambda(x)$ do not differ much. Finally, the following LP optimization is obtained

$$\min_{\tilde{\lambda}} \Bigg\{ G\left(\tilde{\lambda}\right) : \tilde{\lambda}_i \ge 0, \sum_{i=2}^{d_v} \tilde{\lambda}_i = 1, \tilde{p}_e^{(l)} \le p_e^{(l-1)}, \tag{47}$$
$$\max_l \frac{\left|\tilde{p}_e^{(l)} - p_e^{(l)}\right|}{p_e^{(l-1)} - p_e^{(l)}} \le \delta \Bigg\},$$

for $1 \le l \le L$ and $\delta \ll 1$. That is, the LP optimization in (47) searches for a perturbed version of $\lambda(x)$ with better

TABLE II: Optimized variable degree distributions (the code rates are approximately $1/2$).

q	M	Optimized $\lambda(x)$
4	2	$0.0522x + 0.0799x^2 + 0.6983x^3 + 0.1668x^4 + 0.0028x^6$
8	2	$0.0394x + 0.0298x^2 + 0.9007x^3 + 0.0099x^4 + 0.0142x^5 + 0.005x^7 + 0.001x^8$
8	4	$0.0813x + 0.0402x^2 + 0.7717x^3 + 0.0196x^4 + 0.082x^5 + 0.0025x^6 + 0.0007x^7 + 0.002x^9$
16	4	$0.0526x + 0.0584x^2 + 0.8237x^3 + 0.0095x^4 + 0.0104x^5 + 0.044x^6 + 0.0014x^7$
16	8	$0.0226x + 0.2002x^2 + 0.6592x^3 + 0.0593x^4 + 0.0225x^5 + 0.0128x^6 + 0.0091x^7 + 0.0075x^8 + 0.0068x^9$



(a) Union-optimized degree distributions.

(b) BEC LP optimized degree distributions.

Fig. 7: Finite-length decoding performance of the optimized degree distributions for $q = 16, M = 8$ in Table II (rate $1/2$). The BEC LP results are shown for comparison. $\varepsilon_{\rm sh} = 0.66$ is the Shannon limit for these QPEC parameters with rate $1/2$.

performance, i.e., with a smaller number of decoding iterations resulting in the desired probability of decoding failure. The LP optimization is repeated with the optimized $\lambda(x)$ as an input until convergence is achieved.

The major difficulty in solving the LP optimization in (47) in the QPEC case is the need to calculate $p_e^{(l)}$. The calculation of this probability is difficult even when the set cardinality approach is taken, as we saw in Section IV-A. Therefore, we calculate this probability under the union model, motivated by its good approximation of the exact behaviour (see Section V-D). We concentrate on an initial $\rho(x)$ with only two non-zero consecutive degrees, which usually provide good results [19], [26] and limit the search space. To control complexity, the maximum variable node degree is set to $d_v = 10$. The initial $\lambda(x)$ was chosen such that the number of non-zero degrees is small [26] and the initial rate is close to $1/2$. Table II summarizes the optimization results for several values of M and q, where $\rho\left(x\right) = 0.4 \cdot x^6 + 0.6 \cdot x^7$ is the initial check degree distribution.

For comparison, we used the known BEC LP optimization [10] to obtain a good degree distribution with rate $1/2$ (as the code rates in Table II). The BEC optimized variable degree-distribution pair is $\lambda_{\rm BEC}\left(x\right) = 0.3195x + 0.1554x^2 + 0.5251x^9$, obtained by setting $\rho\left(x\right) = 0.4 \cdot x^6 + 0.6 \cdot x^7$ and the BEC threshold 0.473 in the BEC LP. In Figure 7, we present the average finite-length decoding performance of the optimized degree distributions obtained in Table II for $q = 16$ and $M = 8$ compared to the BEC LP optimized degree-distribution pair. As expected, the QPEC-designated optimization (47) provides significantly better results compared to simply using the BEC LP. This is explained by the strict requirement in the BEC LP to recover from full erasures, whereas only $M = 8$ partial erasures should be considered.

VII. Conclusion

This work offers a study of the performance of iterative decoding of GF(q) LDPC codes over the newly defined QPEC. Generalizing the BEC to partial erasures, the QPEC serves as a useful model for partial data loss. We extended the BEC decoder to deal with partial erasures, and demonstrated the spectrum of possible messages in the QPEC decoding process. As a consequence, we showed that the QPEC, unlike the BEC, introduces non-trivial challenges in performance analysis. Therefore, we developed efficient approximation models, which provide important tools for the QPEC decoding performance evaluation. We also suggested LP optimizations for finding LDPC codes with good decoding performance, which provided better results compared to the known BEC LP optimization.

The QPEC model is an initial step in the analysis of measurement channels. These channels and the concept of partial erasures encourage the development of additional models and efficient analysis methods. As a future research, it is suggested to investigate the relation between the algebraic operations of the field and the models proposed in this work. In addition, apart from the random sets provided as the QPEC output, one may consider a model in which the output sets are structured (e.g., contain consecutive levels).

Appendix A
Proof of Theorem 1

Define the sets $\{\mathcal{A}_i\}_{i=1}^{T}$, each containing M elements taken from $\mathcal{X}$, such that $\mathcal{A}_i \neq \mathcal{A}_j$ for $i \neq j$ and $T = \binom{q}{M}$. The output symbol Y is a set of either one symbol or M symbols. Its entropy given an input distribution $\{p_x\}$ (for $x = 0, \alpha^0, \alpha^1, ..., \alpha^{q-2}$) is:

$$H\left(Y;\{p_x\}\right) = -\sum_{x\in\mathcal{X}} p_x(1-\varepsilon)\log\left(p_x(1-\varepsilon)\right)$$
$$-\sum_{i=1}^{T}\left(\frac{\varepsilon}{\binom{q-1}{M-1}}\sum_{x\in\mathcal{A}_i} p_x\right)\log\left(\frac{\varepsilon}{\binom{q-1}{M-1}}\sum_{x\in\mathcal{A}_i} p_x\right). \tag{48}$$

The capacity achieving distribution can be found by solving the following maximization problem:

$$\max_{\{p_x\}} H\left(Y;\{p_x\}\right), \quad \text{s.t.} \sum_{x\in\mathcal{X}} p_x = 1. \tag{49}$$

Using the method of Lagrange multipliers, we get the following system of equations:

$$\frac{\partial H\left(Y;\{p_x\}\right)}{\partial p_x} + \lambda = 0, \text{ for } x = 0, \alpha^0, \alpha^1, ..., \alpha^{q-2} \tag{50}$$
$$\text{s.t.} \sum_{x\in\mathcal{X}} p_x = 1,$$

where λ is the Lagrange multiplier. These equations translate into:

$$-(1-\varepsilon)\left(\log p_x + 1\right) - \sum_{\mathcal{A}_i : x\in\mathcal{A}_i} \frac{\varepsilon}{\binom{q-1}{M-1}} \log\left(\frac{\varepsilon}{\binom{q-1}{M-1}}\sum_{x\in\mathcal{A}_i} p_x\right)$$
$$-\frac{\varepsilon}{\binom{q-1}{M-1}}(T-1) + \lambda = 0, \quad \sum_{x\in\mathcal{X}} p_x = 1, \tag{51}$$

which are satisfied if $p_x = 1/q$ for all $x \in \mathcal{X}$. The mutual information $I\left(X;Y\right)$ is a concave function of p_x, and therefore $p_x = 1/q$ leads to the global maximum of (3), that is, to the capacity. Finally, the QPEC capacity (5) is obtained by substituting (48) and (4) in (3) and setting $p_x = 1/q$ for all x. ∎

Appendix B
Proof of Lemma 9

We begin by proving that $h_\varepsilon(x)$ is an increasing function of both ε and x, for $\varepsilon, x \in [0,1]$, by taking the partial derivatives of $f\left(\varepsilon, x\right)$ with respect to ε and x:

$$\frac{\partial f}{\partial \varepsilon} = \lambda\left(1 - \rho\left(1-x\right) - x\rho'\left(1-x\right)\right), \tag{52}$$

$$\frac{\partial f}{\partial x} = \varepsilon x \lambda'\left(1 - \rho\left(1-x\right) - x\rho'\left(1-x\right)\right)\rho''\left(1-x\right), \tag{53}$$

where $\rho'(x)$ and $\rho''(x)$ denote the first and second derivatives of $\rho(x)$. The polynomials $\rho\left(x\right), \lambda\left(x\right)$ and their derivatives are power series of x with non-negative coefficients, and as such they are non-negative for $x \geq 0$. In particular, $\rho''\left(1-x\right) \geq 0$ since $0 \leq 1-x \leq 1$. Therefore, it is sufficient to prove that $g(x) = \rho\left(1-x\right) + x\rho'\left(1-x\right) \leq 1$ for establishing the non-negativity of the partial derivatives (52) and (53). This is proved in the following manner. The derivative of $g(x)$ in the interval $(0,1)$ satisfies $g'\left(x\right) = -x\rho''\left(1-x\right) < 0$, meaning that $g(x)$ is a decreasing function of x. In particular, $g\left(x\right) \leq g\left(0\right) = \rho\left(1\right) = 1$, as needed. Now, $z_M^{(l)} \geq h_\varepsilon\left(z_M^{(l-1)}\right)$ according to the inequality in (41). Thus,

$$h_\varepsilon\left(z_M^{(l-1)}\right) \geq h_\varepsilon\left(h_\varepsilon\left(z_M^{(l-2)}\right)\right), \quad l \geq 2. \tag{54}$$

As we saw earlier, $h_\varepsilon(x)$ is an increasing function of x. Repeated application of the monotonicity property to the right-hand side of (54) leads to the inequality $h_\varepsilon\left(z_M^{(l-1)}\right) \geq h_\varepsilon^l\left(z_M^{(0)}\right)$, where $h_\varepsilon^l\left(x\right)$ denotes the l^{th} composition of $h_\varepsilon(x)$ with itself. Therefore, $z_M^{(l)} \geq h_\varepsilon\left(z_M^{(l-1)}\right) \geq h_\varepsilon^l\left(z_M^{(0)}\right) = h_\varepsilon^l\left(\varepsilon\right)$, proving the first part of the theorem. The second part of the theorem is proved using the monotonicity property of $h_\varepsilon(x)$ proved above and similar arguments to those used in Section 3.10 of [10] (where the BEC is considered). ∎

References

[1] B. Eitan and A. Roy, "Binary and multilevel flash cells," in *Flash Memories*. Springer US, 1999, pp. 91–152.

[2] E. Gal and S. Toledo, "Algorithms and data structures for flash memories," *ACM Comput. Surv.*, vol. 37, no. 2, pp. 138–163, Jun. 2005.

[3] J. Meena, S. Sze, U. Chand, and T.-Y. Tseng, "Overview of emerging nonvolatile memory technologies," *Nanoscale Research Letters*, vol. 9, no. 1, 2014.

[4] C. Trinh et al., "A 5.6MB/s 64Gb 4b/Cell NAND flash memory in 43nm CMOS," in *IEEE International Solid-State Circuits Conference (ISSCC) 2009*, Feb 2009, pp. 246–247,247a.

[5] X. Huang, M. Asadi, A. Kavcic, and N. Santhanam, "All-bit-line MLC flash memories: Optimal detection strategies," in *2014 IEEE International Conference on Communications (ICC)*, June 2014, pp. 3883–3888.

[6] C. Xu, D. Niu, N. Muralimanohar, N. Jouppi, and Y. Xie, "Understanding the trade-offs in multi-level cell reram memory design," in *Design Automation Conference (DAC)*, May 2013, pp. 1–6.

[7] R. G. Gallager, "Low-density parity-check codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, 1962.

[8] ——, *Low-Density Parity Check Codes*. MIT Press, 1963.

[9] M. Davey and D. MacKay, "Low-density parity check codes over GF(q)," *IEEE Communications Letters*, vol. 2, no. 6, pp. 165–167, 1998.

[10] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.

[11] S.-Y. Chung, J. Forney, G.D., T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the shannon limit," *IEEE Communications Letters*, vol. 5, no. 2, pp. 58–60, Feb 2001.

[12] S. Hongzin and J. R. Cruz, "Reduced-complexity decoding of Q-ary LDPC codes for magnetic recording," *IEEE Transactions on Magnetics*, vol. 39, no. 2, pp. 1081–1087, 2003.

[13] A. Bennatan and D. Burshtein, "Design and analysis of nonbinary LDPC codes for arbitrary discrete-memoryless channels," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 549–583, 2006.

[14] V. Rathi and R. Urbanke, "Density evolution, thresholds and the stability condition for non-binary LDPC codes," *IEE Proceedings-Communications*, vol. 152, no. 6, pp. 1069–1074, Dec 2005.

[15] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 599–618, 2001.

[16] P. Elias, "Coding for two noisy channels," in *Information Theory Third London Symposium*, Sep. 1955, pp. 61–76.

[17] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. IT-27, pp. 533–547, 1981.

[18] T. C. Tao and V. H. Vu, *Additive Combinatorics*. Cambridge University Press, 2006.

[19] T. Richardson, M. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 619–637, 2001.

[20] S. Eliahou and M. Kervaire, "Sumsets in vector spaces over finite fields," *Journal of Number Theory*, vol. 71, no. 1, pp. 12 – 39, 1998.

[21] E. Croot and V. F. Lev, "Open problems in additive combinatorics," Department of Mathematics, Georgia Institute of Technology, Tech. Rep., 2011.

[22] G. Károlyi, "The Cauchy-Davenport theorem in group extensions," *L'Enseignement Mathématique 51*, 2005.

[23] D. J. Grynkiewicz, "On a partition analog of the cauchy-davenport theorem," *Acta Mathematica Hungarica*, vol. 107, no. 1-2, pp. 161–174, 2005.

[24] M. Mitzenmacher and E. Upfal, *Probability and Computing: Randomized algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.

[25] C. M. Grinstead and J. L. Snell, *Introduction to Probability, second edition*. American Mathematical Society, 1997.

[26] S.-Y. Chung, "On the construction of some capacity-approaching coding schemes," Ph.D. dissertation, MIT, 2000.

Rami Cohen (S'12) received the B.Sc. in Electrical Engineering and Physics (*cum laude*) and the M.Sc. in Electrical Engineering from the Technion - Israel Institute of Technology in 2010 and 2012, respectively. He is currently pursuing the Ph.D. degree at the Department of Electrical Engineering, Technion - Israel Institute of Technology. His current research interests lie in coding theory and information theory, in particular the analysis and design of codes for high-speed memory devices and systems.

Yuval Cassuto (S'02-M'08-SM'14) is a faculty member at the Department of Electrical Engineering, Technion – Israel Institute of Technology. His research interests lie at the intersection of the theoretical infomration sciences and the engineering of practical computing and storage systems.

During 2010-2011 he has been a Scientist at EPFL, the Swiss Federal Institute of Technology in Lausanne. From 2008 to 2010 he was a Research Staff Member at Hitachi Global Storage Technologies, San Jose Research Center. From 2000 to 2002, he was with Qualcomm, Israel R&D Center, where he worked on modeling, design and analysis in wireless communications.

He received the B.Sc degree in Electrical Engineering, summa cum laude, from the Technion, Israel Institute of Technology, in 2001, and the MS and Ph.D degrees in Electrical Engineering from the California Institute of Technology, in 2004 and 2008, respectively.

Dr. Cassuto has won the 2010 Best Student Paper Award in data storage from the IEEE Communications Society, as well as the 2001 Texas Instruments DSP and Analog Challenge $100,000 prize.