Distance Verification for Classical and Quantum LDPC Codes

Ilya Dumer, *Fellow, IEEE*, Alexey A. Kovalev, and Leonid P. Pryadko

***Abstract*— The techniques of distance verification known for general linear codes are first applied to the quantum stabilizer codes. Then, these techniques are considered for classical and quantum (stabilizer) low-density-parity-check (LDPC) codes. New complexity bounds for distance verification with provable performance are derived using the average weight spectra of the ensembles of LDPC codes. These bounds are expressed in terms of the erasure-correcting capacity of the corresponding ensemble. We also present a new irreducible-cluster technique that can be applied to any LDPC code and takes advantage of parity-checks' sparsity for both the classical and quantum LDPC codes. This technique reduces complexity exponents of all existing deterministic techniques designed for generic stabilizer codes with small relative distances, which also include all known families of the quantum stabilizer LDPC codes.**

***Index Terms*— Distance verification, quantum stabilizer codes, LDPC codes, erasure correction, list decoding.**

I. Introduction

QUANTUM error correction (QEC) [1]–[3] is a critical part of quantum computing due to the fragility of quantum states. Two related code families, surface (toric) quantum codes [4], [5] and topological color codes [6]–[8], have been of particular interest in quantum design [8], [9]. Firstly, these codes only require simple local gates for quantum syndrome measurements. Secondly, they efficiently correct some non-vanishing fraction of errors, below a fault-tolerant threshold of about 1% per gate. Unfortunately, locality limits such codes to an asymptotically zero code rate [10] and makes a useful quantum computer prohibitively large. Therefore, there is much interest in feasible quantum coding with no local restrictions.

Manuscript received January 29, 2016; revised October 28, 2016; accepted March 13, 2017. Date of publication March 31, 2017; date of current version June 14, 2017. This work was supported in part by NSF under Grant PHY-1415600, in part by the U.S. Army Research Office under Grant W911NF-14-1-0272, and in part by NSF under Grant PHY-1416578. The paper was presented in part at the 2014 IEEE International Symposium on Information Theory and at the 2016 IEEE International Symposium on Information Theory. *(Corresponding Author: Ilya Dumer.)*

I. Dumer is with the Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA 92521 USA (e-mail: dumer@ece.ucr.edu).

A. A. Kovalev is with the Department of Physics and Astronomy, University of Nebraska—Lincoln, Lincoln, NE 68588 USA (e-mail: alexey.kovalev@unl.edu).

L. P. Pryadko is with the Department of Physics and Astronomy, University of California at Riverside, Riverside, CA 92521 USA (e-mail: leonid@ucr.edu).

Communicated by M. Grassl, Associate Editor for Quantum Information Theory.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2017.2690381

Low-density-parity-check (LDPC) codes [11], [12] form a more general class of quantum codes. These codes assume no locality but only require low-weight stabilizer generators (parity checks). Unlike locally-restricted codes, they also achieve a finite code rate along with a non-zero error probability threshold, both in the standard setting, and in a fault-tolerant setting, when syndrome measurements include errors [13], [14]. However, quantum LDPC codes are still much inferior to their classical counterparts. Namely, all existing quantum LDPC codes with bounded stabilizer weight [15]–[23] have code distances d that scale at most as $\sqrt{n \ln n}$ in length n, unlike linear scaling in the classical LDPC codes. Many of the existing quantum constructions also exhibit substantial gaps between the upper and lower bounds for their distances d. In particular, the recent quantum design of [21] yields the orders of n and $\sqrt{n}$ for these bounds. Finding the exact distances of such codes is thus an important open problem.

This paper addresses various numerical algorithms that verify code distance with provable performance for the classical LDPC codes, quantum stabilizer codes, and quantum LDPC codes. Given some ensemble of codes, we wish to verify code distances for most codes in this ensemble with an infinitesimal probability of failure. In particular, we will discuss deterministic algorithms that yield no failures for most codes in a given ensemble. We also address probabilistic algorithms that have a vanishing probability of failure. This high-fidelity setting immediately raises important complexity issues. Indeed, finding the code distance of a generic code is an NP-hard problem. This is valid for both the exact setting [24] and the evaluation problem [25], [26], where we only verify if d belongs to some interval $[\delta, c\delta]$ for a given constant $c \in (1, 2)$. In this regard, we note that all algorithms discussed below still have exponential complexity in block length n, if the average code distance grows linearly in a given ensemble. Below, we consider both binary and q-ary codes and wish to achieve the lowest exponential complexity q^{Fn} for distance verification of classical or quantum LDPC codes.

We analyze complexity exponents F in three steps. Section III establishes a framework for generic quantum codes. To do so, we revisit several algorithms known for classical linear codes. Then we re-apply these techniques for quantum stabilizer codes. Given the complexity benchmarks of Section III, we then address binary LDPC codes in Section IV. Here we can no longer use the generic properties of random generator (or parity-check) matrices. Therefore, we modify the existing algorithms to include the LDPC setting. In particular,

we show that only a vanishing fraction of codes may have atypically high complexity. These codes are then discarded. As a result, we re-define known complexity estimates in terms of two parameters: the average code distance and the erasure-correcting capacity of a specific code ensemble. To estimate this capacity, we use the average weight spectra, which were derived in [27] for the original ensemble of LDPC codes and in [28] for a few other LDPC ensembles. Our complexity estimates hold for any ensemble given its erasure-correcting capacity or some lower bound. More generally, these algorithms perform list decoding within distance d from any received vector y, whereas distance verification does so for $y = 0$.

Here, however, we leave out some efficient algorithms that require more specific estimates. In particular, we do not address belief propagation (BP) algorithms, which can erroneously end when they meet stopping sets, and therefore fail to furnish distance verification with an arbitrarily high likelihood. Despite this, the simulation results presented in papers [29] and [30] show that list decoding BP algorithms can also be effective in distance verification.

In Section V, we consider quantum stabilizer LDPC codes. These codes use some self-orthogonal quaternary code C and its dual $C^{\perp}$. This self-orthogonality separates quantum LDPC codes from their conventional counterparts. One particular difference is a low relative distance of the existing constructions, the other is a substantial number of short cycles in their graphical representation. The latter fact also complicates BP algorithms. For these reasons, our goal is to design new algorithms that are valid for any LDPC code including any quantum code. To do so, we use the fact that verification algorithms may seek only irreducible [14] codewords that cannot be separated into two or more non-overlapping codewords. This approach yields a cluster-based algorithm that exponentially reduces the complexity of all known deterministic techniques for sufficiently small relative distance d/n, which is the case for the existing families of quantum LDPC codes. This algorithm also generalizes the algorithm of [14] for nonbinary LDPC codes.

Consider a q-ary (ℓ, m)-*regular* LDPC code, which has ℓ non-zero symbols in each column and m non-zero symbols in each row of its parity-check matrix. Let $h_2(x)$ be the binary entropy of $x \in [0, 1]$. Our main results are presented in Propositions 7 and 8 and can be summarized as follows.

Proposition 1: Consider any permutation-invariant ensemble $\mathbb{C}$ of q-ary linear codes with relative distance δ_*. Let θ_* denote the expected erasure-correcting capacity for codes $C \in \mathbb{C}$. For most codes $C \in \mathbb{C}$, the code distance $\delta_* n$ can be verified with complexity of order 2^{Fn}, where $F = h_2(\delta_*) - \theta_* h_2(\delta_*/\theta_*)$. For any q-ary (ℓ, m)-regular LDPC code (classical or quantum), the code distance $\delta_* n$ can be verified with complexity of order 2^{Fn}, where $F = \delta_* \log_2(\gamma_m(m-1))$ and γ_m grows monotonically with m in the interval $(1, (q-1)/\ln q)$.

II. Background

Let $C[n, k]_q$ be a q-ary linear code of length n and dimension k in the vector space $\mathbb{F}_q^n$ over the field $\mathbb{F}_q$. This code is specified by the parity check matrix H, namely $C = \{\mathbf{c} \in \mathbb{F}_q^n | H\mathbf{c} = 0\}$. Let d denote the Hamming distance of code C.

A quantum $[[n, k]]$ stabilizer code Q is a 2^k-dimensional subspace of the n-qubit Hilbert space $\mathbb{H}_2^{\otimes n}$, a common $+1$ eigenspace of all operators in an Abelian *stabilizer group* $\mathscr{S} \subset \mathscr{P}_n$, $-\mathbb{1} \notin \mathscr{S}$, where the n-qubit Pauli group $\mathscr{P}_n$ is generated by tensor products of the X and Z single-qubit Pauli operators. The stabilizer is typically specified in terms of its generators, $\mathscr{S} = \langle S_1, \ldots, S_{n-k} \rangle$; measuring the generators S_i produces the *syndrome* vector. The weight of a Pauli operator is the number of qubits it affects. The distance d of a quantum code is the minimum weight of an operator U which commutes with all operators from the stabilizer $\mathscr{S}$, but is not a part of the stabilizer, $U \notin \mathscr{S}$.

A Pauli operator $U \equiv i^m X^{\mathbf{v}} Z^{\mathbf{u}}$, where $\mathbf{v}, \mathbf{u} \in \{0, 1\}^{\otimes n}$ and $X^{\mathbf{v}} = X_1^{v_1} X_2^{v_2} \ldots X_n^{v_n}$, $Z^{\mathbf{u}} = Z_1^{u_1} Z_2^{u_2} \ldots Z_n^{u_n}$, can be mapped, up to a phase, to a quaternary vector, $\mathbf{e} \equiv \mathbf{u} + \omega \mathbf{v}$, where $\omega^2 \equiv \overline{\omega} \equiv \omega + 1$. A product of two quantum operators corresponds to the sum (mod 2) of the corresponding vectors. Two Pauli operators commute if and only if the *trace inner product* $\mathbf{e}_1 * \mathbf{e}_2 \equiv \mathbf{e}_1 \cdot \overline{\mathbf{e}}_2 + \overline{\mathbf{e}}_1 \cdot \mathbf{e}_2$ of the corresponding vectors is zero, where $\overline{\mathbf{e}} \equiv \mathbf{u} + \overline{\omega}\mathbf{v}$. With this map, an $[[n, k]]$ stabilizer code Q is defined by $n - k$ generators of a stabilizer group, which generate some *additive* self-orthogonal code C of size 2^{n-k} over $\mathbb{F}_4$ [31]. The vectors of code C correspond to stabilizer generators that act trivially on the code; these vectors form the *degeneracy group* and are omitted from the distance calculation. For this reason, any stabilizer code Q has a code distance [31] that is defined by the minimum non-zero weight in the code $C^{\perp} \setminus C$.

An LDPC code, quantum or classical, is a code with a sparse parity check matrix. A huge advantage of classical LDPC codes is that they can be decoded in linear time using iterative BP algorithms [32], [33]. Unfortunately, this is not necessarily the case for quantum LDPC codes, which have many short cycles of length four in their Tanner graphs. In turn, these cycles cause a drastic deterioration in the convergence of the BP algorithm [34]. This problem can be circumvented with specially designed quantum codes [19], [35], but a general solution is not known.

III. Generic Techniques for Distance Verification

The problem of verifying the distance d of a linear code (finding a minimum-weight codeword) is related to a more general list decoding problem: find all or some codewords at distance d from the received vector. As mentioned above, the number of operations N required for distance verification can be usually defined by some positive exponent $F = \overline{\lim}\, (\log_q N)/n$ as $n \to \infty$. For a linear q-ary code with k information qubits, one basic decoding algorithm inspects all q^{Rn} distinct codewords, where $R = k/n$ is the code rate. Another basic algorithm stores the list of all q^{n-k} syndromes and coset leaders. This setting gives (space) complexity $F = 1 - R$. We will now survey some techniques that are known to reduce the exponent F for linear codes and re-apply these techniques for quantum codes. For classical codes, most results

discussed below are also extensively covered in the literature (including our citations below). In particular, we refer to [36] for detailed proofs.

A. Sliding Window (SW) Technique

Consider ensemble $\mathbb{C}$ of linear codes $C[n,k]$ generated by the randomly chosen q-ary $(Rn \times n)$ matrices G. It is well known that for $n \to \infty$, most codes in ensemble $\mathbb{C}$ have full dimension $k = Rn$ and meet the asymptotic GV bound $R = 1 - h_q(d/n)$, where

$$h_q(x) = x\log_q(q-1) - x\log_q x - (1-x)\log_q(1-x) \quad (1)$$

is the q-ary entropy function. We use notation c_I and C_I for any vector c and any code C punctured to some subset of positions I. Consider a *sliding window* $I(i,s)$, which is the set of s cyclically consecutive positions beginning with $i = 0,\ldots,n-1$. It is easy to verify that most random q-ary codes $C \in \mathbb{C}$ keep their full dimension Rn on all n subsets $I(i,s)$ of length $s = k + 2\lfloor \log_q n \rfloor$. Let $\mathbb{C}_s$ be such a sub-ensemble of codes $C \in \mathbb{C}$. Most codes $C \in \mathbb{C}_s$ also meet the GV bound, since the remaining codes in $\mathbb{C} \setminus \mathbb{C}_s$ form a vanishing fraction of ensemble $\mathbb{C}$. Also, $\mathbb{C}_s$ includes all cyclic codes. We now consider the following SW technique of [37].

Proposition 2 [37]: The code distance δn of any linear q-ary code $C[n, Rn]$ in the ensemble $\mathbb{C}_s$ can be found with complexity $q^{nF_{\mathrm{C}}}$, where

$$F_{\mathrm{C}} = Rh_q(\delta) \quad (2)$$

For most codes $C \in \mathbb{C}_s$, the complexity exponent is $F^* = R(1-R)$.

Proof: Given a code C, we first verify if $C \in \mathbb{C}_s$, which requires polynomial complexity. For such a code C, consider a codeword $c \in C$ of weight $d = 1,2,\ldots$. The weight of any vector $c_{I(i,s)}$ can change only by one as $i+1$ replaces i. Then some vector $c_{I(i,s)}$ of length s has the average Hamming weight $v \equiv \lfloor ds/n \rfloor$. Consider all

$$L = n(q-1)^v \binom{s}{v}$$

vectors $c_{I(i,s)}$ of weight v on each window $I(i,s)$. Then we use each vector $c_{I(i,s)}$ as an information set and encode it to the full length n. The procedure stops if some encoded vector c has weight d. This gives the overall complexity Ln^2, which has the order of $q^{nF_{\mathrm{C}}}$ of (2). For codes that meet the GV bound, this gives exponent F^*. ∎

Remarks: More generally, the encoding of vectors $c_{I(i,s)}$ represents erasure correction on the remaining $n-s$ positions. We use this fact in Section IV for LDPC codes. Also, any error vector of weight d generates vector u of weight v on some window $I = I(i,s)$. Thus, we can subtract any error vector u from the received vector $y_{I(i,s)}$ and correct d errors in code C.

We now proceed with ensemble $\mathbb{Q}$ of quantum stabilizer codes $\mathcal{Q}$ $[[n, Rn]]$. Most of these codes meet the quantum GV bound [38], [39]

$$R = 1 - 2h_4(\delta) \quad (3)$$

Any code Q is defined by the corresponding additive quaternary code $C^\perp$ and has the minimum distance $d(Q) = d(C^\perp \setminus C)$. Let $\mathbb{Q}_s$ denote the ensemble of codes Q, for which $C^\perp \in \mathbb{C}_s$. Note that $\mathbb{Q}_s$ includes most stabilizer codes.

Corollary 1: The code distance δn of any quantum stabilizer code $Q[[n, Rn]]$ in the ensemble $\mathbb{Q}_s$ can be found with complexity $2^{nF_{\mathrm{SW}}}$, where

$$F_{\mathrm{SW}} = (1+R)h_4(\delta) \quad (4)$$

For most codes in ensemble $\mathbb{Q}_s$, code distances d can be found with the complexity exponent

$$F^*_{\mathrm{SW}} = (1-R^2)/2 \quad (5)$$

Proof: For any quantum stabilizer code $Q[[n,k]]$, we apply the SW procedure to the quaternary code $C^\perp$. Since code C has size 2^{n-k} in the space $\mathbb{F}_4^n$, its dual $C^\perp$ has the effective code rate [1]

$$R' = \left(1 - \frac{n-k}{2n}\right) = (1+R)/2$$

which gives complexity $2^{nF_{\mathrm{SW}}}$ of (4) for a generic stabilizer code Q. Due to possible degeneracy, we also verify that any encoded vector c of weight d does not belong to code C. Most generic codes Q $[[n, Rn]]$ also belong to ensemble $\mathbb{Q}_s$ and therefore satisfy the quantum GV bound. The latter gives exponent (5). ∎

Note that classical exponent $F^* = R(1-R)$ achieves its maximum $1/4$ at $R = 1/2$. By contrast, quantum exponent F^*_{SW} achieves its maximum $1/2$ at the rate $R = 0$.

B. Matching Bipartition (MB) Technique

Proposition 3: The code distance δn of any quantum stabilizer code $Q[[n, Rn]]$ can be found with complexity $2^{nF_{\mathrm{MB}}}$, where

$$F_{\mathrm{MB}} = h_4(\delta). \quad (6)$$

For random stabilizer codes that meet the quantum GV bound (3),

$$F^*_{\mathrm{MB}} = (1-R)/2. \quad (7)$$

Proof: Similarly to the proof of Corollary 1, we consider any stabilizer code Q $[[n, Rn]]$ and the corresponding code $C^\perp$. For code $C^\perp$, we now apply the algorithm of [41] and [42], which uses two similar sliding windows, the "left" window $I_\ell(i, s_\ell)$ of length $s_\ell = \lfloor n/2 \rfloor$ and the complementary "right" window I_r of length $s_r = \lceil n/2 \rceil$. For any vector e of weight d, consider vectors e_ℓ and e_r in windows I_ℓ and I_r. At least one choice of position i then yields the average weights $v_\ell = \lfloor d/2 \rfloor$ and $v_r = \lceil d/2 \rceil$ for both vectors. For each i, both sets $\{e_\ell\}$ and $\{e_r\}$ of such "average-weight" vectors have the size of order $L = (q-1)^{d/2}\binom{n/2}{d/2}$.

We now calculate the syndromes of all vectors in sets $\{e_\ell\}$ and $\{e_r\}$ to find matching vectors (e_ℓ, e_r), which give identical syndromes, and form a codeword. Sorting the elements of the combined set $\{e_\ell\} \cup \{e_r\}$ by syndromes yields all matching pairs

[1] This construction is analogous to pseudogenerators introduced in [40].

with complexity of order $L \log_2 L$. Thus, we find a code vector of weight $d = \delta n$ in any linear q-ary code with complexity of order q^{Fn}, where

$$F = h_q(\delta)/2. \tag{8}$$

For q-ary codes on the GV bound, $F^* = (1 - R)/2$. For stabilizer codes, the arguments used to prove Corollary 1 then give exponents (6) and (7). ∎

Note that the MB-technique works for any linear code, unlike other known techniques provably valid for random codes. For very high rates $R \to 1$, this technique yields the lowest complexity exponent known for classical and quantum codes.

C. Punctured Bipartition (PB) Technique

Proposition 4: The code distance δn of a random quantum stabilizer code $Q[[n, Rn]]$ can be found with complexity $2^{nF_{\rm PB}}$, where

$$F_{\rm PB} = \frac{2(1+R)}{3+R} h_4(\delta) \tag{9}$$

For random stabilizer codes that meet the quantum GV bound (3),

$$F^*_{\rm PB} = (1 - R^2)/(3 + R) \tag{10}$$

Proof: We combine the SW and MB techniques, similarly to the soft-decision decoding of [43]. Let $s = \lceil 2nR/(1+R) \rceil > k$. Then for most random $[n, k]$ codes C, all n punctured codes $C_{I(i,s)}$ are linear random $[s, k]$-codes. Also, any codeword of weight d has average weight $v = \lfloor ds/n \rfloor$ in some window $I(i, s)$. For simplicity, let s and v be even. We then apply the MB technique and consider all vectors e_ℓ and e_r of weight $v/2$ on each window $I(i, s)$. The corresponding sets have the size

$$L_s = (q-1)^{v/2} \binom{s/2}{v/2}.$$

We then select all matching pairs (e_ℓ, e_r) with the same syndrome. The result is the list $\{e\}$ of code vectors of weight v in the punctured $[s, k]$ code $C_{I(i,s)}$. For a random $[s, k]$ code, this list $\{e\}$ has the expected size of order

$$L_v = (q-1)^v \binom{s}{v} / q^{s-k}$$

Each vector of the list $\{e\}$ is re-encoded to the full length n. For each $d = 1, 2, \ldots$, we stop the procedure once we find a re-encoded vector of weight d. The overall complexity has the order of $L_v + L_s$. It is easy to verify [43] that for codes that meet the GV bound, our choice of parameter s gives the same order $L_v \sim L_s$ and minimizes the sum $L_v + L_s$ to the order of $q^{F_* n}$, where

$$F^* = h_q(\delta) R/(1+R) = R(1-R)/(1+R). \tag{11}$$

To proceed with quantum codes $Q[[n, Rn]]$, observe that our parameter s again depends on the effective code rate $R' = (1 + R)/2$. For stabilizer codes, this change yields exponent (9), which gives (10) if codes meet the quantum GV bound. ∎

For codes of rate $R \to 1$ that meet the GV bound, the PB technique gives the lowest known exponents $F^*_{\rm PB}$ (for stabilizer codes) and F^* (for classical q-ary codes). However, no complexity estimates have been proven for specific code families.

Finally, consider the narrower Calderbank-Shor-Steane (CSS) class of quantum codes. Here a parity check matrix is a direct sum $H = G_x \oplus \omega G_z$, and the commutativity condition simplifies to $G_x G_z^T = 0$. A CSS code with $\operatorname{rank} G_x = \operatorname{rank} G_z = (n-k)/2$ has the same effective rate $R' = (1 + R)/2$ since both codes include $k' = n - (n-k)/2 = (n+k)/2$ information bits. Since CSS codes are based on binary codes, their complexity exponents $F(R, \delta)$ can be obtained from (2), (8), and (11) with parameters $q = 2$ and $R' = (1+R)/2$. Here we can also use the GV bound, which reads for CSS codes as follows [44]

$$R = 1 - 2h_2(\delta). \tag{12}$$

D. Covering Set (CS) Technique

This probabilistic technique was proposed in [45] and has become a benchmark in code-based cryptography since classical paper [46]. This technique lowers all three complexity estimates (4), (6), and (9) except for code rates $R \to 1$. The CS technique has also been studied for distance verification of specific code families (see [47], [48]); however, provable results [49], [50] are only known for generic random codes.

Let $C[n, k]$ be some q-ary random linear code with an $r \times n$ parity check matrix H, $r = n - k$. Consider some subset J of $\rho \le r$ positions and the complementary subset I of $g \ge k$ positions. Then the *shortened* code $C_J = \{c_J : c_I = 0\}$ has the parity-check matrix H_J of size $r \times \rho$. We say that matrix H_J has *co-rank* $b(H_J) = \rho - \operatorname{rank} H_J$. Note that $b(H_J) = \dim C_J$, which is the dimension of code C_J.

Proposition 5: The code distance δn of a random quantum stabilizer code $Q[[n, Rn]]$ can be found with complexity $2^{nF_{\rm CS}}$, where

$$F_{\rm CS} = h_2(\delta) - \left(\frac{1-R}{2}\right) h_2\left(\frac{2\delta}{1-R}\right) \tag{13}$$

Proof: First, consider a q-ary code $C[n, k]$. We randomly choose the sets J of r positions to cover every possible set of $d < r$ non-zero positions. To do so, we need no less than

$$T(n, r, d) = \binom{n}{d} \Big/ \binom{r}{d}$$

sets J. On the other hand, the proof of [51, Th. 13.4] shows that a collection of

$$T = T(n, r, d) n \ln n \tag{14}$$

random sets J fails to yield such an (n, r, d)-covering with a probability less than $e^{-n \ln n}$. It is also well known that most $r \times n$ matrices H (excluding a fraction $\binom{n}{r}^{-1}$ of them) yield small co-ranks

$$0 \le b_J \le b_{\max} = \sqrt{2 \log_q \binom{n}{r}} \tag{15}$$

for all square submatrices H_J, $|J| = r$.

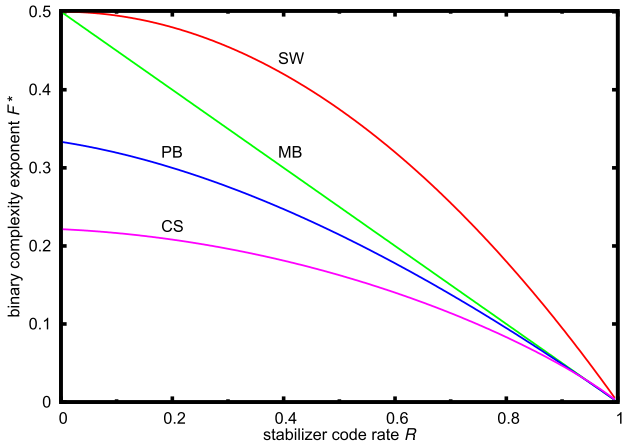

Fig. 1. Complexity exponents of the four generic decoding techniques applied to quantum codes that meet the quantum GV bound (3). SW: sliding window, (5), MB: matching bipartition, (7), PB: punctured bipartition, (10), and CS: covering set, (13).

Given an (n, r, d)-covering W, the CS procedure inspects each set $J \in W$ and discards code C if $\dim C_J > b_{\max}$. Otherwise, it finds the lightest codewords on each set J. To do so, we first perform Gaussian elimination on H_J and obtain a new $r \times r$ matrix $\mathcal{H}_J$ that has the same co-rank $b(H_J)$. Let $\mathcal{H}_J$ include $r - b_J$ unit columns $u_i = (0 \ldots 01_i 0 \ldots 0)$ and b_J other (linearly dependent) columns g_j. All r columns have zeroes in the last b_J positions. If $b_J = 0$ in trial J, then $C_J = 0$ and we proceed further. If $b_J > 0$, the CS algorithm inspects $q^{b_J} - 1$ linear combinations (LC) of columns g_j. Let $LC(p)$ denote some LC that includes p columns g_j. If this $LC(p)$ has weight w, we can nullify it by adding w unit columns u_i and obtain a codeword c of weight $w + p$. The algorithm ends once we find a codeword of weight $w + p = d$, beginning with $d = 2$.

For codes that satisfy condition (15), the CS algorithm has the complexity order of $n^3 q^{b_{\max}} T(n, r, d)$ that is defined by $T(n, r, d)$. For any q, this gives complexity 2^{nF} with exponent

$$F = (1 - R)\big[1 - h_2\left(\delta/(1 - R)\right)\big] \tag{16}$$

For a stabilizer code $[[n, Rn]]$, we obtain (13) using the quaternary code $C^{\perp}$ with the effective code rate $R' = (1 + R)/2$. ■

For stabilizer codes that meet the quantum GV bound (3), exponent F_{CS} of (13) reaches its maximum $F_{\max} \approx 0.22$ at $R = 0$. Their binary counterparts yield exponent (16) that achieves its maximum 0.119 at $R \approx 1/2$.

Discussion: Fig. 1 exhibits different complexity exponents computed for stabilizer codes that meet the quantum GV bound. The CS technique gives the best performance for most code rates $R < 1$, while the two bipartition techniques perform better for high code rates R, which are close to 1. Indeed, equations (7) and (10) scale linearly with $1 - R$, unlike the CS technique that yields a logarithmic slope, according to (13).

More generally, the above algorithms correct the received vector y into the list of codewords located at distance d from y. In this regard, they are similar to list decoding of vector y within a given distance. For example, given an error syndrome $h \neq 0$, MB technique still forms the sets of vectors $\{e_\ell\}$ and $\{e_r\}$. It also derives the syndromes $h(e_\ell)$, but uses the syndromes $h(e_r) + h$ on the right half. Similarly, some SW trials will correctly identify errors on the information blocks and then perform error-free re-encoding. For the CS algorithm, we also make a slight adjustment and inspect all combinations $LC(p) + h$. Each combination $LC(p) + h$ of weight w gives an error of weight $p + w$. It is also important that every trial of the CS algorithm needs only the syndrome h instead of the received vector y. Thus, this algorithm can perform syndrome-based decoding of quantum stabilizer codes.

Finally, let $d_{\mathrm{GV}} = d_{\mathrm{GV}}(n, k)$ be the Gilbert-Varshamov distance of an arbitrary linear $[n, k]$ code. An important result of [37] shows that list decoding of this $[n, k]$ code within a radius d_{GV} gives the block error probability $P(d_{\mathrm{GV}}) \leq 2P_{\mathrm{ML}}$, where P_{ML} is the block error rate of the maximum likelihood decoding for this $[n, k]$ code. Also, note [43] that the block error probability $P(d)$ satisfies inequality $P(d) \leq P_{\mathrm{ML}}(1 + 1/n)$ if $d \geq d_{\mathrm{GV}} + \log_2 n$. Thus, distance verification techniques considered above can be employed for generic list decoding algorithms and maximum likelihood decoding, in particular.

IV. Distance Verification for LDPC Codes

Below, we consider two ensembles of binary (ℓ, m)-LDPC codes with $m \geq \ell \geq 3$. Codes in these ensembles are defined by the binary equiprobable $r \times n$ parity-check matrices H. In ensemble $\mathbb{A}(\ell, m)$, matrices H have all columns of weight ℓ and all rows of weight $m = \ell n/r$. This ensemble also includes a smaller LDPC ensemble $\mathbb{B}(\ell, m)$ originally proposed by Gallager [27]. For each code in $\mathbb{B}(\ell, m)$, its parity-check matrix H is divided into ℓ horizontal blocks $H_1, \ldots, H_\ell$ of size $\frac{r}{\ell} \times n$. Here the first block H_1 consists of m unit matrices of size $\frac{r}{\ell} \times \frac{r}{\ell}$. Any other block H_i is obtained by some random permutation $\pi_i(n)$ of n columns of H_1. Below, we use an equivalent description, where block H_1 also undergoes a random permutation $\pi_1(n)$. Ensembles $\mathbb{A}(\ell, m)$ and $\mathbb{B}(\ell, m)$ have similar spectra and achieve the best asymptotic distance for a given code rate $1 - \ell/m$ among the LDPC ensembles studied to date [28].

For brevity, we say below that a linear code C with N non-zero codewords has *null-free size* N. We also say that code ensemble $\mathbb{C}$ is *permutation-invariant* (PI) if any permutation of positions π in any code $C \in \mathbb{C}$ again gives a code $\pi(C) \in \mathbb{C}$. In particular, LDPC ensembles are in this class. For any subset of positions J of size $\rho = \theta n$, consider all shortened codes $C_J \in \mathbb{C}_J$. Then for any PI ensemble $\mathbb{C}$, all shortened ensembles $\mathbb{C}_J$ have the same expected null-free size N_θ given any J of size θn. By Markov's inequality, for any parameter $t > 0$, at most a fraction $\frac{1}{t}$ of the shortened codes C_J have null-free size exceeding tN_θ on any subset J.

Note that for LDPC codes, parity checks form non-generic sparse matrices H_J. Therefore, below we change the approach of Section III. In essence, we will relate the size 2^{b_J} of codes C_J to the erasure-correcting capacity of LDPC codes. In doing so, we extensively use average weight spectra derived for ensemble $\mathbb{B}(\ell, m)$ in [27] and for ensemble $\mathbb{A}(\ell, m)$ in [28].

This analysis can readily be extended to other ensembles with known average weight spectra. The following results are well known and will be extensively used in our complexity estimates.

Let $\alpha = \ell/m = 1 - R$. For any parameter $\beta \in [0, 1]$, the equation

$$\frac{(1+t)^{m-1} + (1-t)^{m-1}}{(1+t)^m + (1-t)^m} = 1 - \beta \tag{17}$$

has a single positive root t as a function of β. Below we use the parameter

$$q(\alpha,\beta) = \alpha \log_2 \frac{(1+t)^m + (1-t)^m}{2t^{\beta m}} - \alpha m h_2(\beta), \tag{18}$$

where we also take $q(\alpha,\beta) = -\infty$ if m is odd and $\beta \geq 1 - m^{-1}$. Then [28, Th. 4], shows that a given codeword of weight βn belongs to some code in $\mathbb{A}(\ell, m)$ with probability $P(\alpha,\beta)$ such that

$$\lim_{n\to\infty} \frac{1}{n} \log_2 P(\alpha,\beta) = q(\alpha,\beta) \tag{19}$$

Lemma 1: For any given subset J of size θn, where $\theta \leq 1$, codes $C_J(\ell, m)$ of the shortened LDPC ensembles $\mathbb{A}(\ell, m)$ or $\mathbb{B}(\ell, m)$ have the average null-free size N_θ such that

$$\lim_{n\to\infty} \frac{1}{n} \log_2 N_\theta = f(\theta) \tag{20}$$

where

$$f(\theta) = \max_{0<\beta<1} \{q(\alpha, \beta\theta) + \theta h_2(\beta)\} \tag{21}$$

Proof: For any set J of size θn, consider codewords of weight $\beta\theta n$ that have support contained on J. For any $\beta \in (0, 1]$, codes in $\mathbb{A}_J(\ell, m)$ contain the average number

$$N_\theta(\beta) = P(\alpha, \beta\theta) \binom{\theta n}{\beta\theta n} \tag{22}$$

of such codewords of weight $\beta\theta n$. Then

$$\frac{1}{n} \log_2 N_\theta \sim \frac{1}{n} \max_{\beta<1} \log_2 N_\theta(\beta) \sim \max_{\beta<1} \{q(\alpha, \beta\theta) + \theta h_2(\beta)\} \tag{23}$$

which gives asymptotic equalities (20) and (21). ∎

We show in this section that verification complexity is defined by two important parameters, δ_* and θ_*, which are the roots of the equations

$$\begin{aligned} \delta_* : h_2(\delta_*) + q(\alpha, \delta_*) &= 0 \\ \theta_* : f(\theta_*) &= 0. \end{aligned} \tag{24}$$

Discussion: Note that δ_* is the average relative code distance in ensemble $\mathbb{A}(\ell, m)$. Indeed, for $\theta = 1$, equality (22) shows that the average number of codewords $N_\theta(\beta)$ of length n and weight βn has the asymptotic order

$$\frac{1}{n} \log_2 N(\beta) \sim h_2(\beta) + q(\alpha,\beta) \tag{25}$$

Parameter θ_* bounds from below the erasure-correcting capacity of LDPC codes. Indeed, $f(\theta) < 0$ in (21) and $N_\theta = 2^{nf(\theta)} \to 0$ for any $\theta < \theta_*$. Thus, most codes $C \in \mathbb{A}(\ell, m)$ yield only the single-vector codes $C_J(\ell, m) \equiv 0$ and correct any erased set J of size θn. The upper bounds on the erasure-correcting capacity of LDPC codes are also very close to θ_* and we refer to papers [52], [53], where this capacity is discussed in detail.

More generally, consider any PI ensemble $\mathbb{C}$ of q-ary linear codes. We say that θ_* is the *erasure-correcting capacity* for ensemble $\mathbb{C}$ if for any $\epsilon > 0$ the shortened subcodes C_J of length θn, $n \to \infty$, have expected size N_θ such that

$$\begin{cases} N_\theta \to 0, & \text{if } \theta \leq \theta_* - \epsilon \\ N_\theta \geq 1, & \text{if } \theta \geq \theta_* + \epsilon \end{cases} \tag{26}$$

Without ambiguity, we will use the same notation θ_* for any lower bound on the erasure-correcting capacity (26). In this case, we still have asymptotic condition $N_\theta \to 0$ for any $\theta \leq \theta_* - \epsilon$, which is the only condition required for our further estimates. In particular, we use parameter θ_* of (24) for the LDPC ensembles $\mathbb{A}(\ell, m)$ or $\mathbb{B}(\ell, m)$.

For any code rate $R = 1 - \ell/m$, δ_* of (24) falls below the relative GV distance $\delta_{\mathrm{GV}}(R)$ of random codes (see [27], [28]). For example, $\delta_* \sim 0.02$ for the $\mathbb{A}(3, 6)$ LDPC ensemble of rate $R = 1/2$, whereas $\delta_{\mathrm{GV}} \sim 0.11$. On the other hand, θ_* also falls below the erasure-correcting capacity $1 - R$ of random linear codes. For example, $\theta_* = 0.483$ for the ensemble $\mathbb{A}(3, 6)$ of LDPC codes of rate 0.5. In our comparison of LDPC codes and random linear codes, we will show that the smaller distances δ_* reduce the verification complexity for LDPC codes, despite their weaker erasure-correcting capability θ_* for any code rate R.

A. Deterministic Techniques for the LDPC Ensembles.

Proposition 6: Consider any PI ensemble of codes $\mathbb{C}$ with the average relative distance δ_* and the erasure-correcting capacity θ_*. For most codes $C \in \mathbb{C}$, the SW technique performs distance verification with complexity of exponential order q^{Fn} or less, where

$$F = (1 - \theta_*) h_q(\delta_*) \tag{27}$$

Proof: We use the generic SW technique but select sliding windows $I = I(i, s)$ of length $s = (1 - \theta_* + \varepsilon)n$. Here $\varepsilon > 0$ is a parameter such that $\varepsilon \to 0$ as $n \to \infty$. For a given weight $d = \delta_* n$, we again inspect each window $I(i, s)$ and take all L punctured vectors $c_{I(i,s)}$ of average weight $v = \lfloor \delta_* s \rfloor$. Thus,

$$\frac{1}{n} \log_q L \sim \frac{1}{n} \log_q (q-1)^v \binom{s}{v} \sim (1 - \theta_* + \varepsilon) h_q(\delta_*)$$

For each vector $c_{I(i,s)}$, we recover symbols on the complementary set $J = \overline{I}$ of size $(\theta_* - \varepsilon)n$, by correcting erasures in a given code $C \in \mathbb{C}$. This recovery is done by encoding each vector $c_{I(i,s)}$ into C and gives the codeword list of expected size N_θ. Thus, codes C have the average complexity of $n^3 N_\theta L$ combined for all n subsets I. Then only a fraction n^{-1} of such codes may have a complexity above $n^4 N_\theta L$. This gives (27) as $\varepsilon \to 0$. ∎

We proceed with the MB technique, which can be applied to any linear code. For q-ary codes, the MB technique

gives the complexity exponent $F = h_q\ (\delta_*)/2$. Combining Propositions 3 and 6, we have

Corollary 2: Distance verification for most LDPC codes in the ensembles $\mathbb{A}(\ell, m)$ or $\mathbb{B}(\ell, m)$ can be performed with the complexity exponent

$$F = \min\{(1-\theta_*)h_2(\delta_*), h_2(\delta_*)/2\} \tag{28}$$

where parameters δ_* and θ_* are defined in (24).

The PB technique can also be applied to LDPC codes without changes. However, its analysis becomes more involved. Indeed, syndrome-matching in the PB technique yields some punctured (s, k) codes $C_{I(i,s)}$, which are no longer LDPC codes. However, we can still use their weight spectra, which are defined by the original ensemble $\mathbb{C}$ and were derived in [54]. Here we omit lengthy calculations and proceed with a more efficient CS technique.

B. CS Technique for LDPC Ensembles

Below we estimate the complexity of the CS technique for any LDPC code ensemble. Recall from Section III-D that for most linear random $[n, k]$ codes, all shortened codes C_J of length $n-k$ have non-exponential size 2^{b_J}. This is not proven for the LDPC codes or any other ensemble of codes. Therefore, we modify the CS technique to extend it to these non-generic ensembles. In essence, we leave aside the specific structure of parity-check matrices H_J. Instead, we use the fact that atypical codes C_J with large size 2^{b_J} still form a very small fraction of all codes C_J.

Proposition 7: Consider any PI ensemble $\mathbb{C}$ of q-ary linear codes with the average relative distance δ_* and the erasure-correcting capacity θ_*. For most codes $C \in \mathbb{C}$, the CS technique performs distance verification with complexity of exponential order 2^{Fn} or less, where

$$F = h_2(\delta_*) - \theta_* h_2(\delta_*/\theta_*) \tag{29}$$

Proof: We now select sets J of $s = \theta n$ positions, where $\theta = \theta_* - \varepsilon$ and $\varepsilon \to 0$ as $n \to \infty$. To find a codeword of weight d in a given code $C \in \mathbb{C}$, we randomly pick up $T = (n \ln n)\binom{n}{d}/\binom{s}{d}$ sets J. For any J, the shortened code ensemble $\mathbb{C}_J$ has the expected null-free size $N_\theta \to 0$. Let $\mathbb{C}_J(b) \subset \mathbb{C}_J$ be a sub-ensemble of codes $C_J(b)$ that have null-free size $q^b - 1$ for some $b = 0, \dots, \theta n$. Also, let $\alpha_\theta(b)$ be the fraction of codes $C_J(b)$ in the ensemble $\mathbb{C}_J$. Then

$$N_\theta = \sum_{b=0}^{\theta n} \left(q^b - 1\right) \alpha_\theta(b) \tag{30}$$

For each code $C_J(b)$, we again apply Gaussian elimination to its parity-check matrix H_J of size $r \times s$. Similarly to the proof of Proposition 5, we obtain the diagonalized matrix $\mathcal{H}_J$, which consists of $s - b$ unit columns $u_i = (0 \dots 01_i 0 \dots 0)$ and b other columns g_j. To find the lightest codewords on a given set J, we again consider all $q^b - 1$ non-zero linear combinations of b columns g_j. For any given code $C_J(b)$, this gives complexity of order $\mathcal{D}_\theta(i) \le n^3 + rb(q^b - 1) \le n^3(q^b - 1)$. Taking all codes $C_J(b)$ for $b = 0, \dots, \theta n$ on a given set J, we obtain the expected complexity

$$\mathcal{D}_\theta = \sum_{b=0}^{\theta n} n^3(q^b - 1)\alpha_\theta(b) = n^3 N_\theta \tag{31}$$

Thus, the CS algorithm has the expected complexity $\mathcal{D}_{ave} = n^3 T N_\theta$ for all T sets J. Then only a vanishing fraction N_θ/n of codes C have complexity $\mathcal{D} \ge n^4 T$, which gives the exponent $F \le \lim \frac{1}{n} \log_2 \left(n^4 T\right)$ of (29) for most codes. ∎

Discussion: Note that Propositions 6 and 7 employ PI code ensembles $\mathbb{C}$. This allows us to consider all sets J of θn positions and output *all* codewords of weight d for most codes $C \in \mathbb{C}$. If we replace this adversarial model with a less restrictive channel-coding model, we may correct *most* errors of weight d instead of all of them. Then we also remove the above restrictions on ensembles $\mathbb{C}$. Indeed, let us re-define N_θ as the null-free size of codes C_J averaged over all codes $C \in \mathbb{C}$ and all subsets J of size θn. Then we use the following statement:

Lemma 2: Let ensemble $\mathbb{C}$ have vanishing null-free size $N_\theta \to 0$ in the shortened codes C_J of length θn as $n \to \infty$. Then most codes $C \in \mathbb{C}$ correct most erasure subsets J, with the exception of vanishing fraction $\sqrt{N_\theta}$ of codes C and subsets J.

Proof: A code $C \in \mathbb{C}$ fails to correct some erasure set J of weight θn if and only if code C_J has $N_J(C) \ge 1$ non-zero codewords. Let M_θ be the average fraction of such codes C_J taken over all codes C and all subsets J. Note that $M_\theta \le N_\theta$. Per Markov's inequality, no more than a fraction $\sqrt{M_\theta}$ of codes C may leave a fraction $\sqrt{M_\theta}$ of sets J uncorrected. ∎

Finally, we summarize the complexity estimates for classical binary LDPC codes in Fig. 2. For comparison, we also plot two generic exponents valid for most linear binary codes. The first exponent

$$F = \min\{R(1-R), (1-R)/2\} \tag{32}$$

combines the SW and MB algorithms, and the second exponent (16) represents the CS algorithm. For LDPC codes,

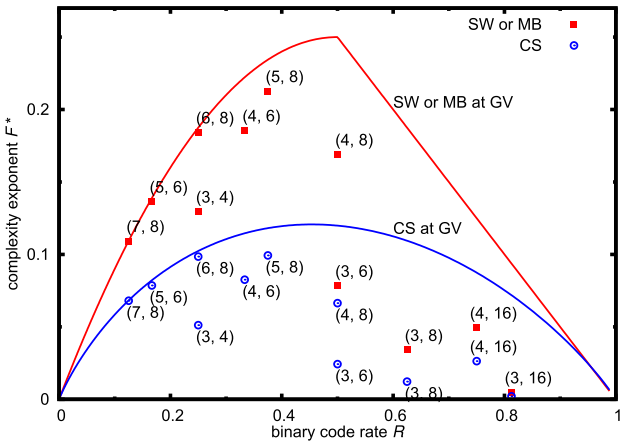

Fig. 2. Complexity exponents for the binary codes meeting the GV bound and for some (ℓ, m)-regular LDPC codes as indicated. "SW or MB" stands for deterministic techniques from Eq. (28) for LDPC codes, or Eq. (32) for codes meeting the GV bound, and CS stands for covering set technique, Eq. (29) for LDPC codes, or Eq. (16) for codes meeting the GV bound.

we similarly consider the exponent (28) that combines the SW and MB algorithms and the exponent (29) that represents the CS algorithm for the LDPC codes. Here we consider ensembles $\mathbb{A}(\ell, m)$ or $\mathbb{B}(\ell, m)$ for various LDPC (ℓ, m) codes with code rates ranging from 0.125 to 0.8. With the exception of low-rate codes, all LDPC codes of Fig. 2 have substantially lower distances than their generic counterparts. This is the reason LDPC codes also achieve an exponentially smaller complexity of distance verification despite their lower erasure-correcting capacity.

V. Irreducible-Cluster (IC) Technique

The complexity estimates of Sec. IV rely on the average weight distributions of binary (ℓ, m)-regular LDPC codes and hold for most codes in the corresponding ensembles. Here we suggest a deterministic distance-verification technique, which is applicable to any q-ary (ℓ, m)-regular LDPC code, quantum or classical. First, we define irreducible codewords.

Definition 1: Given a linear q-ary code $\mathcal{C}_q$, we say that a codeword c is irreducible if it cannot be represented as a linear combination of two codewords with non-overlapping supports.

Our technique is based on the following simple lemma.

Lemma 3 [14]: A minimum-weight codeword of a linear code $\mathcal{C}_q$ is irreducible.

IC Algorithm: General Description: Let a q-ary (ℓ, m)-regular LDPC code be defined by a list $\mathcal{L}$ of parity checks b with supports J_b of size m. The following algorithm finds an irreducible codeword c of weight d. The algorithm performs multiple runs and includes a variable number $\omega \leq d-1$ of steps in each run. The initial step $i=0$ of each run is given a position $j_0 = 0, \ldots, n-1$ and the symbol $c_{j_0} = 1$. The input to each consecutive step i includes some previously derived sub-vector $c(J_i)$ with its support J_i. It also includes the ordered sublist $\mathcal{N}_i \subset \mathcal{L}$ of all parity checks b unsatisfied by sub-vector $c(J_i)$. Then step i extends vector $c(J_i)$ with some non-overlapping subset $c(I_i)$ of v_i new non-zero symbols. The extension $I_i, c(I_i)$ is chosen to make the first parity check $b^{(1)} \in \mathcal{N}_i$ satisfied on the extended support $J_{i+1} = J_i \cup I_i$:

$$\sum_{j \in J_i} b_j^{(1)} c_j + \sum_{j \in I_i} b_j^{(1)} c_j = 0 \tag{33}$$

The result is the extended vector $c(J_{i+1})$ and the new list $\mathcal{N}_{i+1}$ of parity checks unsatisfied by $c(J_{i+1})$. Clearly, $\mathcal{N}_{i+1}$ excludes parity check $b^{(1)}$. It may also drop some other checks in $\mathcal{N}_i$, which were satisfied in step i, but may include new parity checks, which become unsatisfied due to the newly added symbols. Note that a parity check dropped in step i may later re-appear in some list $\mathcal{N}_s$, $s > i+1$. Each run must satisfy restrictions (33) for all steps and end with d symbols, thus

$$\sum_{i=1}^{\omega} v_i = d-1 \tag{34}$$

Each run ends with a complete selection list $\{I_i, c(I_i) \,|\, i = 0, \ldots, \omega\}$ and gives a codeword of weight d if the list $\mathcal{N}_{\omega+1}$ is empty. For a quantum stabilizer code, we also verify the restriction $c \in C^{\perp} \setminus C$. Given no codeword of weight d, we proceed with a new run, which employs a new selection list. We will now limit possible choices of all vectors $c(I_i)$.

Additively Irreducible Selection: We say that a new selection $I, c(I)$ of non-zero symbols is *additively irreducible (AI)* for a parity-check b if any non-empty subset $I' \subset I$ satisfies restriction

$$\sum_{j \in I'} b_j c_j \neq 0 \tag{35}$$

From now on, any selection list $\{I_i, c(I_i) \,|\, i = 0, \ldots, \omega\}$ must also satisfy restrictions (35) in each step i. We proceed with the following observations.

A. If an AI vector satisfies parity check $b^{(1)}$, then no smaller subset $c(I')$ can do so. Indeed, let restrictions (33) hold on the sets I and $I' \subset I$. Then we obtain equality $\sum b_j c_j = 0$ on the subset $I \setminus I'$, which contradicts (35). We also see that for any reducible vector $c(I)$ that satisfies the current check $b^{(1)}$, there exists its sub-vector $c(I')$, which also satisfies $b^{(1)}$.

B. We may process parity checks one-by-one. Indeed, irrespective of the order in which parity checks are processed, the codewords will satisfy all parity checks after w steps. We may also set $c_{j_0} = 1$ in a linear code C. Our brute-force algorithm begins with a correct choice of j_0 for some runs and then exhausts all possible irreducible selections. Thus, in each step, one of the runs begins with a correct subvector $c(J_i)$ and then adds some correct AI subvector $c(I_i)$.

C. The algorithm may terminate only at some codeword of weight d. More generally, the algorithm can return all (non-collinear) irreducible vectors up to some weight D.

D. If some run fails in step w, we can return to step $w-1$ and exhaust all choices of vectors $c(I_{w-1})$. Similarly, we can return to step $w-2$ and so on. This back-and-forth version slightly reduces the overall complexity; however, it will keep its asymptotic order.

Let $N_v(q, b)$ denote the number of q-ary vectors $c(I)$ of length v that satisfy restrictions (33) and (35). Clearly,

$$N_v(q, b) \leq (q-1)^{v-1} \tag{36}$$

Below, we use notation $N_v(q)$ since we will prove that all parity checks b give the same number $N_v(q, b) \equiv N_v(q)$. Note also that the AI restriction (35) drastically limits the number $N_v(q)$ for small q. For example, a binary parity check $b^{(1)}$ is satisfied in (33) only if v is odd; however, any string of $v \geq 3$ ones includes a subset of two ones and contradicts the AI property (35). Thus, $v = 1$ for $q = 2$ and $N_1(2) = 1$.

We now proceed with complexity estimates. First, we employ a trivial upper bound (36). We further reduce this number in Lemma 4.

Let $\delta_{a,b}$ be the Kronecker symbol, $h = d-1$ and $t = m-1$. Recall that each run is defined by some set $\{I_i, c(I_i) \,|\, i = 0, \ldots, \omega\}$. Given restriction (34), the number of runs is bounded from above by the quantities

$$S_h(m, q) \equiv \sum_{\omega \geq 1} \sum_{v_i \in \{1,2,\ldots,t\}} \delta_{h, v_1 + \ldots + v_\omega} \prod_{i=1}^{\omega} N_{v_i}(q) \binom{t}{v_i} \tag{37}$$

which have the power-moment generating function

$$g(z) = 1 + \sum_{h=1}^{\infty} S_h(m,q) z^h = \sum_{\omega=0}^{\infty} [T(z)]^{\omega} = [1 - T(z)]^{-1}, \quad (38)$$

$$T(z) \equiv \sum_{h=1}^{t} z^h N_h(q) \binom{t}{h}. \quad (39)$$

We can now derive the coefficients $S_h(m,q)$. This can be done by the Chernoff bound, similarly to the estimates of [27] or by the combinatorial technique of [28]. Instead, we use another simple technique that employs contour integration and gives the exact formulas for the coefficients $S_h(m,q)$ along with their exponential orders. Namely, let the denominator $1-T(z)$ in (38) have $s \le t$ distinct roots z_r, $r = 0, 1, \ldots, s-1$, with ordered magnitudes $\rho = |z_0| \le |z_1| \le \ldots \le |z_{s-1}|$. Then coefficients $S_h(m,q)$ can be derived by a contour integration over a circle of radius $\epsilon < \rho$ around the origin,

$$S_h(m,q) = \frac{1}{2\pi i} \oint \frac{dz}{z^d} \frac{1}{1-T(z)} = -\sum_{r=0}^{s-1} \mathrm{Res}\left(\frac{1}{z^d [1-T(z)]}, z_r \right) \quad (40)$$

where $\mathrm{Res}(f(z), a)$ is the residue of $f(z)$ at a. For large weights d, the exponential order of $S_h(m,q)$ is defined by the root z_0, which has the smallest magnitude ρ. Next, note that $z_0 = \rho > 0$ is strictly positive and non-degenerate, since the coefficients of $T(z)$ are non-negative. In this case,

$$\mathrm{Res}\left(\frac{1}{z^d [1-T(z)]}, z_0 \right) = -\frac{1}{z_0^d T'(z_0)} \quad (41)$$

where $T'(z)$ is the derivative of the polynomial $T(z)$; it is non-negative at $z = z_0$. This gives the exponential bound

$$S_h(m,q) \le c\rho^{-d} + \mathcal{O}(|z_1|^{-d}) \sim c[\gamma_m (m-1)]^d \quad (42)$$

with the complexity exponent $\gamma_m \equiv 1/[(m-1)\rho]$.

We now employ upper bound (36). In this case, equality (39) gives the polynomial

$$\overline{T}(z) = \frac{1}{q-1} \left\{ [(q-1)z + 1]^t - 1 \right\}$$

which has the roots

$$z_r = (q^{1/t} e^{2\pi i r/t} - 1)/(q-1), \quad r = 0, 1, \ldots, t-1$$

Thus, the asymptotic expansion (42) yields the constant

$$c = \frac{1 + (q-1)\rho}{qt} = \frac{q^{1/(m-1)}}{q(m-1)}$$

and the complexity exponent

$$\overline{\gamma}_m = \frac{q-1}{(m-1)\left(q^{1/(m-1)} - 1\right)} \le \overline{\gamma}_{\infty} = \frac{q-1}{\ln q} \quad (43)$$

As a side remark, note that the larger values $\upsilon_i > 1$ reduce the number of terms in the product taken in (37); therefore, they contribute relatively little to the overall sum $S_h(m,q)$. It is for this reason that a simple bound (36) can yield a reasonably tight estimate (43). Our next step is to reduce the exponent $\overline{\gamma}_m$ by limiting the number $N_{\upsilon}(q,b)$. Let $M_{\upsilon}(q)$ denote the set of q-ary vectors $c(I)$ of length υ that satisfy the restrictions

$$\sum_{j \in I'} c_j \ne 0 \text{ for all } I' \subseteq I \quad (44)$$

Let $A_{\upsilon}(q)$ be the size of $M_{\upsilon}(q)$ and $\upsilon_{\max}$ be the maximum length of vectors in $M_{\upsilon}(q)$.

Lemma 4: The number $N_{\upsilon}(q,b)$ of q-ary vectors $c(I)$ of length υ, which satisfy restrictions (33) and (35) in a Galois field F_q, does not depend on a parity check b and is equal to $A_{\upsilon}(q)/(q-1)$. For any $q = 2^u$, $\upsilon_{\max} = u$ and $N_{\upsilon}(q) = (q-2)\cdot\ldots\cdot(q-2^{\upsilon-1})$. For a prime number q, $\upsilon_{\max} = q-1$.

Proof: Let two sets of q-ary vectors $c(I,b)$ and $c(I,B)$ of length υ satisfy restrictions (33) and (35) for some parity checks b and B. Then any such vector $c(I,B)$ has its counterpart $c(I,b)$ with symbols $c_j(I,b) = B_j c_j(I,b)/b_j$. Thus, the two sets have the same size and $N_{\upsilon}(q,b) = N_{\upsilon}(q)$. We can also specify AI-restrictions (35) using AI-restrictions (44) for the parity check $b^* = (1, \ldots, 1)$ and all subsets $I' \subset I$. Now let $\lambda \ne 0$ be the value of the first summand in (33) for some unsatisfied parity check. Consider a subset of vectors in $M_{\upsilon}(q)$ that satisfy restriction $\sum_I c_j = -\lambda$. This subset has the size $A_{\upsilon}(q)/(q-1)$ and satisfies both restrictions (33) and (35) for the parity check b^*. Thus, $N_{\upsilon}(q) = A_{\upsilon}(q)/(q-1)$.

Next, consider the Galois field F_q for $q = 2^u$. Then the sums in the left-hand side of (44) represent all possible linear combinations over F_2 generated by υ or fewer elements of $M_{\upsilon}(q)$. Thus, any symbol $c_j(I)$ must differ from the linear combinations of the previous symbols $c_1(I), \ldots, c_{j-1}(I)$. This gives the size $A_{\upsilon}(q) = (q-1)(q-2)\cdot\ldots\cdot(q-2^{\upsilon-1})$ and also proves that $\upsilon_{\max} = u$.

For any prime number q, any sum of s elements in (44) must differ from the sums of $t < s$ elements on its subsets.

TABLE I

NUMBER OF ADDITIVELY-IRREDUCIBLE q-ARY STRINGS OF LENGTH υ FOR $q = p^m$

υ	$q=2$	$q=3$	$q=4$	$q=5$	$q=8$
1	1	1	1	1	1
2	0	1	2	3	6
3		0	0	4	24
4				1	0

TABLE II

COEFFICIENT γ_m OF THE COMPLEXITY EXPONENT $\delta \log_q(\gamma_m(m-1))$ FOR DIFFERENT m AND q

m	$q=2$	$q=3$	$q=4$	$q=5$	$q=8$
3	1	1.2071	1.3660	1.5	1.8229
5	1	1.2906	1.5	1.7331	2.2779
10	1	1.3333	1.5672	1.8555	2.5051
10^2	1	1.3631	1.6135	1.9416	2.6626
10^3	1	1.3657	1.6176	1.9493	2.6765
∞	1	1.3660	1.6180	1.9501	2.6780
Upper bound $(q-1)/\ln q$	1.4427	1.8205	2.1640	2.4853	3.3663

Thus, different sums may take at most $\upsilon_{\max}$ non-zero values for $s = 1, \ldots, \upsilon_{\max}$ and $\upsilon_{\max} \leq q-1$. Then $\upsilon_{\max} = q-1$ is achieved on the vector $c = (1, \ldots, 1)$ of length $q-1$. ■

Lemma 4 shows that the numbers $N_v(q)$ and the lengths $\upsilon_{\max}$ differ substantially for different q. Some of these quantities are listed in Table I for small q. Table II gives some exponents γ_m obtained for irreducible clusters, along with the upper bound $\overline{\gamma}_\infty$ (valid for all clusters) in the last row. We summarize our complexity estimates as follows.

Proposition 8: A codeword of weight δn in any q-ary (ℓ, m) LDPC code can be found with complexity $2^{F_{\rm IC}n}$, where

$$F_{\rm IC} = \delta \log_2(\gamma_m(m-1)),$$

$\gamma_m \in (1, \gamma_\infty)$ grows monotonically with m and $\gamma_\infty < \overline{\gamma}_\infty = (q-1)/\ln q$.

Remarks: The algorithm presented here for linear q-ary codes generalizes an algorithm described in [14] for binary codes. It can be also applied to a more general class of q-ary (ℓ, m)-limited LDPC codes, whose parity check matrices have all columns and rows of Hamming weights no more than ℓ and m, respectively. This algorithm is also valid for q-ary CSS codes, and gives the same complexity exponent. However, for q-ary stabilizer codes, the numbers of additively irreducible clusters (e.g., from Table I) have to be increased by an additional factor of q^v, $N_v^{(\rm stab)}(q) = q^v N_v(q)$. As a result, the complexity exponents in Table II also increase, $\gamma_m^{(\rm stab)} = q\gamma_m$. In particular, for qubit stabilizer codes, $q = 2$, we obtain complexity exponent $\gamma_m^{(\rm qubit)} = 2$.

Also, note that for the existing quantum LDPC codes with distance d of order $\sqrt{n}$, the presented IC algorithm has the lowest proven complexity among deterministic algorithms. Indeed, exponent $F_{\rm IC}$ is linear in the relative distance δ, whereas deterministic techniques of Sec. III give the higher exponents $F \to \delta \log(1/\delta)$ in this limit. In this regard, exponent $F_{\rm IC}$ performs similarly to the CS exponent $F_{\rm CS}$ of generic codes, which is bounded by $\delta - \delta \log_2(1-R)$ and is linear in δ.

VI. Further Extensions

In this paper, we study provable algorithms of distance verification for LDPC codes. More generally, this approach can be used for any ensemble of codes with a given relative distance δ_* and erasure-correcting capacity θ_*.

One particular extension is any ensemble of irregular LDPC codes with known parameters δ_* and θ_*. Note that parameter θ_* has been studied for both ML decoding and message-passing decoding of irregular codes [52], [53], [55]. For ML decoding, this parameter can also be derived using the weight spectra obtained for irregular codes in papers [56], [57]. Also, these techniques can be extended to ensembles of q-ary LDPC codes. The weight spectra of some q-ary ensembles are derived in [58] and [59].

Another direction is to design more advanced algorithms of distance verification for LDPC codes. Most of such algorithms known to date for linear $[n, k]$ codes combine the MB and CS techniques. In particular, algorithm [60] takes a linear $[n, k]$-code and seeks some high-rate punctured $[k+\mu, k]$-block that has $\varepsilon \ll k$ errors among k information bits and μ error-free parity bits. The search is conducted similarly to the CS technique. Then the MB technique corrects ε errors in this high-rate $[k+\mu, k]$-code. A slightly more efficient algorithm [61] simplifies this procedure and seeks punctured $[k+\mu, k]$-code that has $\varepsilon \ll k+\mu$ errors spread across information and parity bits. In this case, the optimal choice of parameters ε and μ reduces the maximum complexity exponent $F(R)$ to 0.1163. Later, this algorithm was re-established in [62] and [63], with detailed applications for the McEliece cryptosystem. More recently, the maximum complexity exponent $F(R)$ has been further reduced to 0.1019 using some robust MB techniques that allow randomly overlapping partitions [64]. An important observation is that both the MB and CS techniques can be applied to LDPC codes; therefore, our conjecture is that provable complexity bounds for distance verification also carry over to these more advanced techniques.

Acknowledgment

The authors would like to thank the reviewers for helpful remarks.

References

[1] P. W. Shor, "Scheme for reducing decoherence in quantum computer memory," *Phys. Rev. A, Gen. Phys.*, vol. 52, p. R2493, Oct. 1995. [Online]. Available: http://link.aps.org/abstract/PRA/v52/pR2493

[2] E. Knill and R. Laflamme, "Theory of quantum error-correcting codes," *Phys. Rev. A, Gen. Phys.*, vol. 55, no. 2, pp. 900–911, Feb. 1997. [Online]. Available: http://dx.doi.org/10.1103/PhysRevA.55.900

[3] C. H. Bennett, D. DiVincenzo, J. A. Smolin, and W. K. Wootters, "Mixed-state entanglement and quantum error correction," *Phys. Rev. A, Gen. Phys.*, vol. 54, p. 3824, Nov. 1996. [Online]. Available: http://dx.doi.org/10.1103/PhysRevA.54.3824

[4] A. Y. Kitaev, "Fault-tolerant quantum computation by anyons," *Ann. Phys.*, vol. 303, pp. 2–30, Jan. 2003. [Online]. Available: http://arxiv.org/abs/quant-ph/9707021

[5] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, "Topological quantum memory," *J. Math. Phys.*, vol. 43, no. 9, p. 4452, 2002. [Online]. Available: http://dx.doi.org/10.1063/1.1499754

[6] H. Bombin and M. A. Martin-Delgado, "Topological quantum distillation," *Phys. Rev. Lett.*, vol. 97, no. 18, p. 180501, Oct. 2006. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevLett.97.180501

[7] H. Bombin and M. A. Martin-Delgado, "Optimal resources for topological two-dimensional stabilizer codes: Comparative study," *Phys. Rev. A, Gen. Phys.*, vol. 76, no. 1, p. 012305, Jul. 2007.

[8] H. Bombin and M. A. Martin-Delgado, "Homological error correction: Classical and quantum codes," *J. Math. Phys.*, vol. 48, no. 5, p. 052105, 2007. [Online]. Available: http://scitation.aip.org/content/aip/journal/jmp/48/5/10.1063/1.2731356

[9] R. Raussendorf and J. Harrington, "Fault-tolerant quantum computation with high threshold in two dimensions," *Phys. Rev. Lett.*, vol. 98, p. 190504, May 2007. [Online]. Available: http://link.aps.org/abstract/PRL/v98/e190504

[10] S. Bravyi, D. Poulin, and B. Terhal, "Tradeoffs for reliable quantum information storage in 2D systems," *Phys. Rev. Lett.*, vol. 104, p. 050503, Feb. 2010. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevLett.104.050503

[11] M. S. Postol. (2001). "A proposed quantum low density parity check code." [Online]. Available: http://arxiv.org/abs/quant-ph/0108131

[12] D. J. C. MacKay, G. Mitchison, and P. L. McFadden, "Sparse-graph codes for quantum error correction," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2315–2330, Oct. 2004. [Online]. Available: http://dx.doi.org/10.1109/TIT.2004.834737

[13] A. A. Kovalev and L. P. Pryadko, "Fault tolerance of quantum low-density parity check codes with sublinear distance scaling," *Phys. Rev. A, Gen. Phys.*, vol. 87, p. 020304(R), Feb. 2013. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevA.87.020304

[14] I. Dumer, A. A. Kovalev, and L. P. Pryadko, "Thresholds for correcting errors, erasures, and faulty syndrome measurements in degenerate quantum codes," *Phys. Rev. Lett.*, vol. 115, p. 050502, Jul. 2015. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevLett.115.050502

[15] J.-P. Tillich and G. Zemor, "Quantum LDPC codes with positive rate and minimum distance proportional to $n^{1/2}$," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2009, pp. 799–803.

[16] G. Zémor, "On Cayley graphs, surface codes, and the limits of homological coding for quantum error correction," in *Proc. 2nd Int. Workshop Coding Cryptol. (IWCC)*, 2009, pp. 259–273. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-01877-0_21

[17] A. Couvreur, N. Delfosse, and G. Zémor, "A construction of quantum LDPC codes from Cayley graphs," *IEEE Trans. Inf. Theory*, vol. 59, no. 1, pp. 6087–6098, Sep. 2013.

[18] A. A. Kovalev and L. P. Pryadko, "Improved quantum hypergraph-product LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2012, pp. 348–352.

[19] I. Andriyanova, D. Maurice, and J.-P. Tillich, "New constructions of CSS codes obtained by moving to higher alphabets," to be published. [Online]. Available: https://arxiv.org/pdf/1202.3338

[20] A. A. Kovalev and L. P. Pryadko, "Quantum Kronecker sum-product low-density parity-check codes with finite rate," *Phys. Rev. A*, vol. 88, p. 012311, Jul. 2013. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevA.88.012311

[21] S. Bravyi and M. B. Hastings, "Homological product codes," in *Proc. 46th Annu. ACM Symp. Theory Comput. (STOC)*, New York, NY, USA, May/Jun. 2014, pp. 273–282.

[22] L. Guth and A. Lubotzky, "Quantum error correcting codes and 4-dimensional arithmetic hyperbolic manifolds," *J. Math. Phys.*, vol. 55, no. 8, p. 082202, 2014. [Online]. Available: http://scitation.aip.org/content/aip/journal/jmp/55/8/10.1063/1.4891487

[23] M. H. Freedman, D. A. Meyer, and F. Luo, "Z_2-systolic freedom and quantum codes," in *Computational Mathematics*. London, U.K.: Chapman & Hall, Feb. 2002, pp. 287–320. [Online]. Available: http://dx.doi.org/10.1201/9781420035377.ch12

[24] A. Vardy, "The intractability of computing the minimum distance of a code," *IEEE Trans. Inf. Theory*, vol. 43, no. 6, pp. 1757–1766, Nov. 1997.

[25] I. Dumer, D. Micciancio, and M. Sudan, "Hardness of approximating the minimum distance of a linear code," *IEEE Trans. Inf. Theory*, vol. 49, no. 1, pp. 22–37, Jan. 2003.

[26] Q. Cheng and D. Wan, "A deterministic reduction for the gap minimum distance problem," in *Proc. STOC*, 2009, pp. 33–38.

[27] R. G. Gallager, *Low Density Parity Check Codes*. Cambridge, MA, USA: MIT Press, 1963.

[28] S. Litsyn and V. Shevelev, "On ensembles of low-density parity-check codes: Asymptotic distance distributions," *IEEE Trans. Inf. Theory*, vol. 48, no. 4, pp. 887–908, Apr. 2002.

[29] D. Declercq and M. Fossorier, "Improved impulse method to evaluate the low weight profile of sparse binary linear codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2008, pp. 1963–1967.

[30] X.-Y. Hu, M. P. C. Fossorier, and E. Eleftheriou, "Approximate algorithms for computing the minimum distance of low-density parity-check codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun./Jul. 2004, p. 475.

[31] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, "Quantum error correction via codes over GF(4)," *IEEE Trans. Inf. Theory*, vol. 44, no. 4, pp. 1369–1387, Jul. 1998. [Online]. Available: http://dx.doi.org/10.1109/18.681315

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

[33] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms*. New York, NY, USA: Cambridge Univ. Press, 2003. [Online]. Available: http://www.cs.toronto.edu/~mackay/itila/p0.html

[34] D. Poulin and Y. Chung, "On the iterative decoding of sparse quantum codes," *Quant. Inf. Comput.*, vol. 8, no. 10, p. 987, 2008.

[35] K. Kasai, M. Hagiwara, H. Imai, and K. Sakaniwa, "Quantum error correction beyond the bounded distance decoding limit," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 1223–1230, Feb. 2012.

[36] A. Barg, "Complexity issues in coding theory," in *Handbook of Coding Theory*, V. Pless and W. C. Huffman, Eds. Amsterdam, The Netherlands: Elsevier, 1998, pp. 649–754.

[37] G. S. Evseev, "Complexity of decoding for linear codes," (in Russian), *Problemy Peredachi Informacii*, vol. 19, no. 1, pp. 3–8, 1983. [Online]. Available: http://mi.mathnet.ru/eng/ppi1159

[38] A. Ekert and C. Macchiavello, "Quantum error correction for communication," *Phys. Rev. Lett.*, vol. 77, pp. 2585–2588, Sep. 1996. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevLett.77.2585

[39] K. Feng and Z. Ma, "A finite Gilbert–Varshamov bound for pure stabilizer quantum codes," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3323–3325, Dec. 2004. [Online]. Available: http://dx.doi.org/10.1109/TIT.2004.838088

[40] G. White and M. Grassl, "A new minimum weight algorithm for additive codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2006, pp. 1119–1123.

[41] I. I. Dumer, "On syndrome decoding of linear codes," (in Russian), in *Proc. 9th All-Union Symp. Redundancy Inf. Syst.*, vol. 2. May 1986, pp. 157–159.

[42] I. I. Dumer, "Two decoding algorithms for linear codes," (in Russian), *Problemy Peredachi Informacii*, vol. 25, no. 1, pp. 24–32, 1989. [Online]. Available: http://mi.mathnet.ru/eng/ppi635

[43] I. Dumer, "Soft-decision decoding using punctured codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 1, pp. 59–71, Jan. 2001.

[44] A. R. Calderbank and P. W. Shor, "Good quantum error-correcting codes exist," *Phys. Rev. A*, vol. 54, no. 2, pp. 1098–1105, Aug. 1996.

[45] E. Prange, "The use of information sets in decoding cyclic codes," *IRE Trans. Inf. Theory*, vol. 8, no. 5, pp. 5–9, Sep. 1962.

[46] R. McEliece, "A public-key cryptosystem based on algebraic coding theory," JPL, Pasadena, CA, USA, DSN Prog. Rep. 43-44, 1978.

[47] P. J. Lee and E. F. Brickell, "An observation on the security of McEliece's public-key cryptosystem," in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science), vol. 330, D. Barstow *et al.*, Eds. Berlin, Germany: Springer, 1988, pp. 275–280.

[48] J. Leon, "A probabilistic algorithm for computing minimum weights of large error-correcting codes," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 1354–1359, Sep. 1988.

[49] E. A. Kruk, "Decoding complexity bound for linear block codes," (in Russian), *Problemy Peredachi Informatsii*, vol. 25, no. 3, pp. 103–107, 1989. [Online]. Available: http://mi.mathnet.ru/eng/ppi665

[50] J. T. Coffey and R. M. Goodman, "The complexity of information set decoding," *IEEE Trans. Inf. Theory*, vol. 36, no. 5, pp. 1031–1037, Sep. 1990.

[51] P. Erdos and J. Spencer, *Probabilistic Methods in Combinatorics*. Budapest, Hungary: Akademiai Kiado, 1974.

[52] T. L. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

[53] H. Pishro-Nik and F. Fekri, "On decoding of low-density parity-check codes over the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 50, no. 3, pp. 439–454, Mar. 2004.

[54] C.-H. Hsu and A. Anastasopoulos, "Capacity achieving LDPC codes through puncturing," *IEEE Trans. Inf. Theory*, vol. 54, no. 10, pp. 4698–4706, Oct. 2008.

[55] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.

[56] C. Di, R. Urbanke, and T. Richardson, "Weight distributions: How deviant can you be?" in *Proc. Int. Symp. Inf. Theory (ISIT)*, Washington, DC, USA, Jun. 2001, p. 50.

[57] S. Litsyn and V. Shevelev, "Distance distributions in ensembles of irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 49, no. 12, pp. 3140–3159, Dec. 2003.

[58] L. Andriyanova, V. Rathi, and J.-P. Tillich, "Binary weight distribution of non-binary LDPC codes," in *Proc. Int. Symp. Inf. Theory (ISIT)*, Seoul, South Korea, 2009, pp. 65–69.

[59] S. Yang, T. Honold, Y. Chen, Z. Zhang, and P. Qiu, "Weight distributions of regular low-density parity-check codes over finite fields," *IEEE Trans. Inf. Theory*, vol. 57, no. 11, pp. 7507–7521, Nov. 2011.

[60] J. Stern, "A method for finding codewords of small weight," in *Coding Theory and Applications* (Lecture Notes in Computer Science), vol. 388, G. Cohen and J. Wolfmann, Eds. Heidelberg, Germany: Springer, 1989, pp. 106–113.

[61] I. Dumer, "On minimum distance decoding of linear codes," in *Proc. 5th Sov.-Swedish Int. Workshop Inf. Theory*, Moscow, Russia, Jan. 1991, pp. 50–52.

[62] M. Finiasz and N. Sendrier, "Security bounds for the design of code-based cryptosystems," in *Advances in Cryptology—ASIACRYPT* (Lecture Notes in Computer Science), vol. 5912, M. Matsui, Ed. Heidelberg, Germany: Springer, 2009, pp. 88–105.

[63] D. J. Bernstein, T. Lange, and C. Peters, "Smaller decoding exponents: Ball-collision decoding," in *Advances in Cryptology—CRYPTO*, vol. 6841, P. Rogaway, Ed. Heidelberg, Germany: Springer, 2011, pp. 743–760.

[64] A. Becker, A. Joux, A. May, and A. Meurer, "Decoding random binary linear codes in $2^{n/20}$: How $1 + 1 = 0$ improves information set decoding," in *Advances in Cryptology—EUROCRYPT*, vol. 7237, D. Pointcheval and T. Johansson, Eds. Heidelberg, Germany: Springer, 2012, pp. 520–536.

Ilya Dumer (M'94–SM'04–F'07) received the M.Sc. degree from the Moscow Institute of Physics and Technology, Russia, in 1976 and the Ph.D. degree from the Institute for Information Transmission Problems of the Russian Academy of Sciences, in 1981. From 1983 to 1995, he was with the Institute for Information Transmission Problems. Since 1995, he has been a Professor of Electrical Engineering at the University of California, Riverside. During 1992-1993, he was a Royal Society Guest Research Fellow at Manchester University, Manchester, U.K., and during 1993-1994, an Alexander von Humboldt Fellow at the Institute for Experimental Mathematics in Essen, Germany. His research interests are in coding theory, discrete geometry, and their applications. During 2006-2009, Dr. Dumer served as an Associate Editor for the IEEE TRANSACTIONS ON INFORMATION THEORY.

Alexey Kovalev received the M.Sc. degree from the Moscow Institute of Physics and Technology, Russia, in 1999 and the Ph.D. degree from Delft University of Technology, in 2006. Since 2013, he has been an Assistant Professor in the Department of Physics and Astronomy at the University of Nebraska-Lincoln. His research interests are in condensed matter physics and quantum information.

Leonid P. Pryadko received the M.Sc. degree from the Moscow Institute of Physics and Technology in 1988 and the Ph.D. degree from Stanford University in 1996. Since 2000, he has been with the department of Physics and Astronomy at the University of California, Riverside, where he is currently a Professor. His research interests are in condensed matter physics, statistical mechanics, and quantum information.