Title
Distance verification for LDPC codes

Permalink
https://escholarship.org/uc/item/4x6487jf

Authors
Dumer, Ilya
Kovalev, Alexey A
Pryadko, Leonid P
et al.

Publication Date
2016

DOI
10.1109/isit.2016.7541755

Peer reviewed
Distance verification for LDPC codes

Ilya Dumer*, Alexey A. Kovalev†, and Leonid P. Pryadko‡

* ECE Department, University of California at Riverside, USA (e-mail: dumere@ee.ucr.edu)
† Department of Physics & Astronomy, University of Nebraska-Lincoln, USA (e-mail: alexey.kovalev@unl.edu)
‡ Department of Physics & Astronomy, University of California at Riverside, USA (e-mail: leonid@ucr.edu)

Abstract—The problem of finding code distance has been long studied for the generic ensembles of linear codes and led to several algorithms that substantially reduce exponential complexity of this task. However, no asymptotic complexity bounds are known for distance verification in other ensembles of linear codes. Our goal is to re-design the existing generic algorithms of distance verification and derive their complexity for LDPC codes. We obtain new complexity bounds with provable performance expressed in terms of the erasure-correcting thresholds of long LDPC codes. These bounds exponentially reduce complexity estimates known for linear codes.

Index Terms—Distance verification, complexity bounds, LDPC codes, erasure correction, covering sets

I. INTRODUCTION

This paper addresses the problem of finding code distances of LDPC codes with provable complexity estimates. Note that finding code distance \( d \) of a generic code is an NP-hard problem. This is valid for both the exact setting [1] and the evaluation problem [2], [3], where we only verify if an arbitrarily high level of accuracy.

Below we address generic algorithms of distance verification - known for linear codes - and re-design these algorithms for LDPC codes. The main problem is that such algorithms heavily rely on the properties of the randomly chosen generator (or parity-check) matrices. These properties have not been proved (or do not hold) for the smaller ensembles of codes, such as cyclic codes, LDPC codes, and others. Therefore, we will use a different technique and derive complexity estimates for LDPC codes using a single parameter, which is the erasure-correcting threshold of a specific code ensemble. We then define this threshold via the average weight spectra of LDPC codes. This technique is different from the generic approach.

In particular, we calculate the average complexity of distance verification and then discard a vanishing fraction of LDPC codes that have atypically high complexity. Our main result is the new complexity bounds for distance verification for ensembles of LDPC codes or other ensembles with a given erasure-correcting threshold. These algorithms perform with an arbitrarily high likelihood. Some other algorithms also include impulse techniques [4] that apply list decoding BP algorithms to the randomly induced errors. Simulation results presented in [4] show that impulse techniques can also be effective in distance verification albeit with a lesser fidelity.

II. BACKGROUND

Let \( C[n, k] \) be a linear binary code of length \( n \) and dimension \( k \). The problem of verifying distance \( d \) of a linear code (finding a minimum-weight codeword) is related to the decoding problem: find an error of minimum weight that gives the same syndrome as the received codeword. The number of operations \( N \) required for distance verification can usually be defined by some positive exponent \( F = \lim \left( \log_2 N / n \right) \) as \( n \to \infty \). For example, for any code \( C[n, k] \), inspection of all \( 2^k \) distinct codewords has (time) complexity exponent \( F = R \), where \( R = k / n \) is the code rate. Given substantially large memory, one can instead consider the syndrome table that stores the list of all \( q' \) syndromes and coset leaders, where \( r = n - k \). This setting gives (space) complexity \( F = 1 - R \).

To proceed with the more efficient algorithms, we also need to consider some parameters of the shortened and punctured codes. Let \( G \) and \( H \) denote a generator and parity check matrices of a code \( C[n, k] \). Let \( I \) be some subset of \( g \geq k \) positions and \( J \) be the complementary subset of \( \eta = n - g \) positions. Consider the punctured code \( C_I = \{ c_I : c \in C \} \) generated by submatrix \( G_I \) of size \( k \times g \). The complementary shortened code \( C_J = \{ c_J : c_I = 0 \} \) has parity-check matrix \( H_J \) of size \( r \times \eta \). These matrices include at most \( k \) and \( \eta \) linearly independent rows, respectively. Let \( b(G_I) = k - \text{rank } G_I \) and \( b(H_J) = \eta - \text{rank } H_J \) denote the co-ranks of these two matrices. Throughout the paper, we use the following simple statement.

Lemma 1. For any linear code \( C[n, k] \), matrices \( G_I \) and \( H_J \) have equal co-ranks \( b(G_I) = b(H_J) \) on the complementary subsets \( I \) and \( J \).

Proof: Code \( C_J \) has size \( 2^\text{dim } C_J = 2^{\text{rank } H_J} = 2^b(H_J) \) and contains all (shortened) code vectors \( c \) with \( c_I = 0 \). On the other hand, for a given matrix \( G_I \), there are \( 2^b(G_I) \) vectors \( c \) with \( c_I = 0 \). Thus, \( b(G_I) = b(H_J) \).

Now consider the shortened codes \( C_J \) of length \( \eta = \theta n \) taken over different sets \( J \). Let codes \( C_J \) have the average size \( N_\theta = 2^\text{dim } C_J \). Then Markov’s inequality gives another useful estimate.
Corollary 1. For any subset $J$ and any $t > 0$, at most a fraction $\frac{1}{t}$ of the shortened codes $C_J$ have size exceeding $tN_0$.

We will now consider two ensembles of regular LDPC codes. Ensemble $h_0(\ell, m)$ is defined by the equiprobable $r \times n$ matrices $H$ that have all columns of weight $\ell$ and all rows of weight $m = \ell n / r$. Below we take $m \geq \ell \geq 3$. This ensemble also includes a smaller LDPC ensemble $\mathbb{B}(\ell, m)$ originally proposed by Gallager [5]. For each code in $\mathbb{B}(\ell, m)$, its parity-check matrix $H$ consists of $\ell$ horizontal blocks $H_1, \ldots, H_\ell$ of size $\frac{\ell}{\ell} \times n$. The first block $H_1$ consists of $m$ consecutive unit matrices of size $\frac{\ell}{\ell} \times \frac{\ell}{\ell}$. Any other block $H_1$ is obtained by some random permutation $\pi_1(n)$ of $n$ columns of $H_1$. Ensembles $h_0(\ell, m)$ and $\mathbb{B}(\ell, m)$ have similar spectra and achieve the best asymptotic distance for a given code rate $1 - \ell / m$ among various LDPC ensembles studied to date [6].

Note that LDPC codes are defined by non-generic, sparse parity check matrices $H_J$. Below we will relate the co-ranks $b_J = b(H_J)$ of these matrices $H_J$ to the erasure-correcting thresholds of LDPC codes. In doing so, we extensively use the average weight spectra derived for the ensemble $\mathbb{B}(\ell, m)$ in [5] and for ensemble $h_0(\ell, m)$ in [6]. We note, however, that this analysis can readily be extended to other ensembles with the known average weight spectra.

Let $\alpha = \ell / m = 1 - R$. For any parameter $\beta \in [0, 1]$, consider the equation
\[
\frac{(1 + t)^{m-1}}{1 + (1 - t)^{m-1}} = 1 - \beta \tag{1}
\]
that has a single positive root $t$. Also, let $h(\beta)$ be the binary entropy function. Below we extensively use the parameter
\[
q(\alpha, \beta) = \alpha \log_2 \frac{(1 + t)^m + (1 - t)^m}{2\beta^m} - \alpha h(\beta), \tag{2}
\]
and also take $q(\alpha, \beta) = -\infty$ if $m$ is odd and $\beta \geq 1 - \frac{1}{m}$. Then Theorem 4 of [6] shows that a given codeword of weight $\beta n$ belongs to some code in ensemble $h_0(\ell, m)$ with probability $P(\alpha, \beta)$ such that
\[
\lim_{n \to \infty} \frac{1}{n} \log_2 P(\alpha, \beta) = q(\alpha, \beta) \tag{3}
\]

Lemma 2. For any given subset $J$ of size $\theta n$, where $\theta \leq 1$, codes $C_J(\ell, m)$ of the shortened ensemble $h_0(\ell, m)$ have the average number $N_0$ of nonzero codewords such that
\[
\lim_{n \to \infty} \frac{1}{n} \log_2 N_0 = f(\theta) \tag{4}
\]
\[
f(\theta) := \max_{0 < \beta \leq 1} \{ q(\alpha, \beta \theta) + \theta h(\beta) \} \tag{5}
\]

Proof: For any set $J$ of size $\theta n$, consider codewords $c$ of weight $\beta n$ that have support on $J$. For any $\beta \in (0, 1]$, codes in $h_0(\ell, m)$ contain the average number $N_0(\beta) = P(\alpha, \beta\theta)$ of such codewords $c$. Then
\[
\max_{0 < \beta \leq 1} \frac{1}{n} \log_2 N_0(\beta) \sim \max_{0 < \beta \leq 1} \{ q(\alpha, \beta \theta) + \theta h(\beta) \} \tag{6}
\]
which gives asymptotic equalities (4) and (5).

III. DISTANCE VERIFICATION FOR LDPC CODES

A. Two main parameters for complexity estimates.

Two essential differences separate LDPC ensembles from random codes in regards to complexity estimates. These differences are closely related to two parameters, $\delta_*$ and $\theta_*$, which are the roots of the equations
\[
\delta_* : h(\delta_*) + q(\alpha, \delta_*) = 0 \tag{7}
\]
\[
\theta_* : f(\theta_*) = 0. \tag{7}
\]
Note that $\delta_*$ is the average relative code distance in the ensemble $h_0(\ell, m)$. Indeed, for $\theta = 1$, equality (6) shows that the average number of codewords $N_0(\beta)$ of length $n$ and weight $\beta n$ has asymptotic order
\[
\frac{1}{n} \log_2 N(\beta) \sim h(\beta) + q(\alpha, \beta) \tag{8}
\]
For any code rate $R = 1 - \ell / m$, $\delta_*$ falls below the GV distance $h^{-1}(1 - R)$ of random codes (see [5] and [6] for more details). For example, $\delta_* \approx 0.02$ for the $h(3, 6)$ ensemble of rate $R = 1/2$, whereas $h^{-1}(0.5) \approx 0.11$. The smaller distances $\delta_*$ will reduce the complexity of distance verification.

Parameter $\theta_*$ also plays a significant role in distance verification. Namely, consider a code ensemble $C$ of growing length $n \to \infty$. Let $N_0$ be the number of nonzero codewords in the shortened codes $C_J$ averaged over all codes $C \in C$ and all subsets $J$ of size $\theta n$. Then we use the following statement.

Lemma 3. Let the ensemble $C$ have a vanishing average number $N_0 \to 0$ of nonzero codewords in the shortened codes $C_J$ of length $\theta n$. Then most codes $C \in C$ correct most erasure subsets $J$, with the exception of a vanishing fraction of codes $C$ and subsets $J$.

Proof: A code $C \in C$ fails to correct some erasure set $J$ of weight $\theta n$ iff code $C_J$ has $N_J(C) \geq 1$ nonzero codewords. Let $M_0$ be the average fraction of such codes $C_J$ taken over all codes $C$ and all subsets $J$. Note that $M_0 \leq N_0$. Per Markov’s inequality, no more than a fraction $\sqrt{M_0}$ of codes $C$ may leave a fraction $\sqrt{M_0}$ of sets $J$ uncorrected.

More generally, we say that an ensemble of codes $C$ has the erasure-correcting threshold $\theta_* \in (0, 1]$ if $N_0 \to 0$ for any $\theta < \theta_*$ and $N_0 \geq 1$ for any $\theta > \theta_*$ on the sets $J$ of size $\theta n$. Here ensembles $h_0(\ell, m)$ and $\mathbb{B}(\ell, m)$ satisfy Lemma 3 for any $\theta < \theta_*$ of (7). Thus, $\theta_*$ serves as a lower bound on the erasure-correcting capacity of LDPC codes under ML decoding. Alternatively, one can use other thresholds, such as the threshold for message-passing algorithms. Note that ensembles $h_0(\ell, m)$ and $\mathbb{B}(\ell, m)$ are permutation-invariant and therefore yield the same fraction of uncorrected codes $C$ for each erasure subset $J$. Then for any $\varepsilon > 0$, the bound $N_J(C) \leq 2^{n \varepsilon}$ holds on all subsets $J$ (except for a fraction of $2^{-\varepsilon n}$ of codes $C$) as long as $N_0 \leq 1$.

For LDPC codes, $\theta_* < \alpha$, where $\alpha = 1 - R$ is the erasure-correcting threshold for random linear codes. For example, $\theta_* = 0.483$ for the ensemble $h(3, 6)$ of LDPC codes. See also papers [7]–[10], where parameter $\theta_*$ is discussed in a greater detail for both ML decoding and message-passing decoder.
The reduced erasure-correcting threshold \( \theta_s \) will increase complexity estimates for LDPC codes. In the sequel, we will show that the first factor (the smaller distance \( \delta_s \)) outweighs the second factor (the smaller threshold \( \theta_s \)) and reduces complexity of distance verification for LDPC codes.

B. Sliding window (SW) technique for LDPC codes

This technique of [11] decodes generic linear codes \( C[n, k, d] \) generated by the randomly chosen \((Rn \times n)\) matrices \( G \). Note that most such codes have full dimension \( k = Rn \) and meet the asymptotic GV bound \( d \). It is shown in [11] that nearly full decoding (that has error probability similar to that of ML decoding) can be performed for most codes \( C[n, k, d] \) with complexity of order \( 2^{nR(1-R)} \).

Below we modify this algorithm for other ensembles of codes, such as \( A(\ell, m) \) or \( B(\ell, m) \).

**Proposition 1.** Consider any ensemble of codes \( C \) with an average relative distance \( \delta_s \) and an erasure-correcting bound \( \theta_s \). For most codes \( C \in C \), SW technique performs distance verification with complexity of exponential order \( 2^{Fn} \) or less, where

\[
F = (1 - \theta_s)h(\delta_s) \tag{9}
\]

**Proof:** Consider a sliding window \( I(i,s) \), which is the set of \( s \) cyclically consecutive positions for some \( i = 0, \ldots, n-1 \). We choose \( s = (1 - \theta_s + \epsilon)n \), where \( \epsilon > 0 \) is a parameter such that \( \epsilon \to 0 \) as \( n \to \infty \). A window \( I(i,s) \) can change its Hamming weight only by one when it moves from position \( i \) to \( i + 1 \); thus any codeword \( v \) of weight \( d = \delta_s n \) has at least one window \( I(i,s) \) with the average Hamming weight \( v = [\delta_s n] \). For each window \( I \), we inspect all \( L = \binom{n}{s} \) vectors \( c_L \) of weight \( v \). Here

\[
\frac{1}{n} \log_2 L = (1 - \theta_s + \epsilon)h(\delta_s)
\]

We then encode each vector \( c_L \) performing erasure correction on the complementary sets \( J \) of size \( (\theta_s - \epsilon)n \). Thus, a typical vector \( c_L \) generates the average number \( N_\theta \) of nonzero codewords \( c_J \). Given \( L \) vectors \( c_L \) and \( n \) sets \( I = I(i,s) \), we obtain the average encoding complexity of \( n^3 N_\theta L \). Here we take the average over different codes \( C \in C \). Thus, at most a vanishing fraction \( n^{-3} \) of such codes \( C \) have complexity above \( n^3 N_\theta L \) for all \( n \) subsets \( J \). This gives (9) as \( \epsilon \to 0 \). \( \square \)

C. Matching Bipartition (MB) technique for LDPC codes

Below we briefly discuss MB-technique of [12], [13]. It works for any linear code and yields the lowest asymptotic complexity for very high code rates \( R \to 1 \).

**Proposition 2.** MB technique performs distance verification for a linear code of distance \( \delta_s n \) with complexity of exponential order \( 2^{Fn} \), where

\[
F = h(\delta_s)/2 \tag{10}
\]

**Proof:** To find an (unknown) vector \( e \) of weight \( d = \delta_s n \), we use the “left” window \( I_L \) of length \( s_L = [n/2] \) starting in any position \( i \) and the complementary “right” window \( I_R \) of length \( s_r = [n/2] \). At least one choice of \( i \) gives the average weights \( v_L = [d/2] \) and \( v_r = [d/2] \) for truncated vectors \( e_L \) and \( e_r \), in windows \( I_L \) and \( I_r \). The number \( L \) of vectors \( e_L \) and \( e_r \) has the order of

\[
\frac{1}{n} \log_2 L \sim \frac{1}{n} \log_2 \binom{n}{s} \sim h(\delta_s)/2
\]

We calculate the syndromes of all vectors \( e_L \) and \( e_r \) and try to match two vectors with equal syndromes. This matching is performed by sorting the elements of the combined set with complexity of order \( L \log_2 L \), which gives exponent (10). \( \square \)

Exponents (9) and (10) give the combined estimate

\[
F = \min\{(1 - \theta_s)h(\delta_s), h(\delta_s)/2\} \tag{11}
\]

Here parameters \( \delta_s \) and \( \theta_s \) are defined for LDPC codes in (7).

D. Covering set (CS) technique for LDPC codes

This probabilistic technique was proposed in [14] and has become a benchmark in cryptography since the classical paper [15]. It lowers complexity estimate (11) for all but very high code rates \( R \to 1 \). CS technique has also been studied for distance verification of specific code families (see [16] and [17]); however, provable results [18], [19] are only known for generic random codes.

Below we choose any LDPC ensemble and describe CS technique in the following proposition.

**Proposition 3.** Consider any code ensemble \( C \) with an average relative distance \( \delta_s \) and an erasure-correcting bound \( \theta_s \). For most codes \( C \in C \), CS technique performs distance verification or corrects up to \( \delta_s n \) errors with complexity of order \( 2^{Fn} \) or less, where

\[
F = h(\delta_s) - \theta_s h(\delta_s/\theta_s) \tag{12}
\]

**Proof:** Let \( e \) be some unknown codeword of weight \( d \) in a given code \( C \in C \). Alternatively, we can consider an error vector \( e \) of weight \( d \). To find \( e \), we repeatedly try to cover all \( d \) nonzero positions of \( e \) with some subsets \( J = \{i_1, \ldots, i_s\} \) of \( s = \theta n \) positions, \( \theta = \theta_s - \epsilon \) and \( \epsilon \to 0 \) as \( n \to \infty \). To cover every possible \( d \)-set, we need no less than

\[
T(n, s, d) = \binom{n}{d}/\binom{s}{d} \tag{13}
\]

sets \( J \). Below we randomly choose a larger number of

\[
T = T(n, s, d) n \ln n
\]

sets \( J \). Following Theorem 13.4 of [20] it is easy to see that \( T \) trials fail to yield such an \((n, s, d)\)-covering with a probability less than \( e^{-n \ln n} \).

Recall that \( N_\theta \to 0 \) for the shortened codes \( C_J \). Let \( C_J(b) \) be a code that contains \( 2^b - 1 \) nonzero codewords for some \( b = 0, \ldots, \theta n \). Also, let \( \alpha(b) \) be the fraction of codes \( C_J(b) \) in the ensemble \( C_J \). Then

\[
N_\theta = \sum_{b=0}^{\theta n} (2^b - 1) \alpha(b) \tag{14}
\]

A parity-check matrix \( H_I \) of any code \( C_J(b) \) has rank \( s - b \) and size \( r \times s \), where \( r = n - k \) is the number of parity
checks. By Gaussian elimination, matrix $H_J$ can be modified into a new $r \times s$ matrix $H_J'$ that includes $b$ zero rows. We will also place $s-b$ unit columns $u_i = (0,0\ldots,0,1,0\ldots,0)$ in the first positions $i \in [1, s-b]$ of $H_J'$, and $b$ other columns $g_j$ in the last positions $j \in [s-b+1, s]$. Let $v = He^T$ denote the syndrome of vector $e$ (possibly modified by the Gaussian elimination procedure).

A. First, consider general error correction given a syndrome $v \neq 0$. If $b = 0$ in a given trial $J$, then matrix $H_J$ has full rank and we obtain vector $e$ of weight $w(t)$ if $b > 0$, we assume that $v$ contains only zero symbols in the last $b$ positions. Then CS algorithm inspects all $2^b$ linear combinations (LC) of the last columns $g_j$. Let $LC(p)$ denote some LC that includes $p$ columns. If $LC(p) + v$ has weight $w$, we obtain vector $e$ of weight $w + p$ by adding $w$ unit columns $u_i$.

The overall decoding algorithm successively tries to find a vector $e$ of weight $d = 1, 2,\ldots$. For any given $d$, it runs over all subsets $J$ and ends once we find a vector $e$ of weight $w + p = d$. For any given code $C_J(b)$, this procedure includes one Gaussian elimination and up to $b$ vector additions, which gives complexity $D(b) \leq n^3 + rb2^b \leq n^32^b$. For a given set $J$, different codes $C_J(b)$ yield the average complexity

$$D(b) = \sum_{b=0}^{\hat{b}_m} n^32^b \alpha(b) = n^3N_\theta + n^3$$

Thus, CS algorithm has the total average complexity $D_{ave} \sim n^{3}T$ for all $T$ sets $J$. Then at most a vanishing fraction $1/n$ of codes $C$ have complexity $D \geq n^{3}T$, which gives the exponent (12) for the remaining codes in ensemble $\mathcal{C}$ as $n \rightarrow \infty$.

B. Vector $e$ forms a codeword with syndrome $v = 0$. Then any code $C_J(0)$ has no nonzero codewords, and CS algorithm skips the above case $b = 0$. Also, we consider only $2^b - 1$ nonzero combinations $LC(p)$ for the last $b$ columns in any $C_J(b)$. Thus, we replace (15) with a similar inequality

$$D(b) \leq \sum_{b=1}^{\hat{b}_m} n^3 (2^b - 1) \alpha(b) \leq n^3N_\theta + n^3$$

that satisfies complexity bound (12).

**Remark.** The existing CS algorithms employ some stringent properties of random ensembles of linear codes. For example, the algorithm of [18] uses the fact that most random binary $r \times n$ matrices $H$, except an exponentially small fraction $\binom{n}{r}^{1-c}$ for $c > 1$, have all $r \times r$ submatrices $H_J$ with nearly-full rank $r - b$, where

$$0 \leq b \leq b_{max} = \sqrt{c \log_2 \binom{n}{r}}$$

Thus, all shortened codes $C_J$ have limited size $2^b$ for most linear codes $C$. For LDPC codes, we use a slightly weaker condition. Our technique discards codes $C_J$ of large size $2^b$ that form an exponentially small fraction of all codes $C_J$.

Fig. 1 summarizes complexity estimates for LDPC codes. For comparison, we also plot two generic exponents valid for most linear codes. Note that these codes meet the GV bound and have parameters $h(\delta_*) = \theta_* = 1 - R$. Then the combination (11) of SW and MB algorithms gives exponent $F = \min\{R(1 - R), (1 - R)/2\}$, whereas exponent (12) of CS algorithm reads as $F = (1 - R)[1 - \delta/(1 - R)]$. For LDPC codes, we similarly consider the exponents (11) and (12). Here we consider ensembles $k(\ell, m)$ or $B(\ell, m)$ for various LDPC $(\ell, m)$ codes with code rates ranging from 0.125 to 0.8. With the exception of low-rate codes, all LDPC codes of Fig. 1 achieve a substantial reduction in complexity exponent for distance verification compared to the generic linear codes.

![Complexity exponents for the binary codes meeting the GV bound and for some $(\ell, m)$-regular LDPC codes as indicated. Abbreviation “SW or MB” stands for the Sliding Window or Matching Bipartition techniques (marked with filled boxes), and “CS” stands for the Covering-Set technique (marked with empty circles).](image)

### IV. FURTHER EXTENSIONS

In this paper, we study provable algorithms of distance verification for LDPC codes and derive complexity estimates using only the relative distance $\delta_*$ and the erasure-correcting threshold $\theta_*$ averaged over a given ensemble of codes. For LDPC codes, these algorithms exponentially reduce generic complexity estimates known for random linear codes. More generally, this approach can be used for any ensemble of codes with a given erasure-correcting threshold.

One particular extension is any ensemble of irregular LDPC codes with the known parameters $\delta_*$ and $\theta_*$. Note that parameter $\theta_*$ has been studied for both ML decoding and message-passing decoding of irregular codes [7], [8], [10]. For ML decoding, this parameter can also be derived using the weight spectra obtained for irregular codes in papers [21], [22].

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 Matching Bipartition (MB) techniques with the Covering Set (CS) algorithms. In particular, the algorithm of [23] first applies CS technique seeking some slightly corrupted information set of $k$ bits. It also tries to select some small subset of $\Delta$ parity bits, every time assuming that these bits are error-free. Then MB technique is applied to correct information bits in the
$[k + \Delta, k]$-code with $\Delta$ correct parity bits. This algorithm reduces the maximum complexity exponent $\max_{R} \mathcal{F}(R)$ of CS technique from 0.1208 to 0.1167. A slightly more efficient approach is by NSF Grant PHY-1415600.

LDPC codes that require stringent self-orthogonality conditions. These more advanced algorithms can again slightly reduce the complexity exponent $\mathcal{F}(R)$ distance verification also carry over to the above techniques. Therefore our conjecture is that provable complexity bounds for this approach requires substantial improvements as complexity exponents exceed the exponent $\mathcal{F}(R)$ obtained in the current paper.

Acknowledgment. The work of L.P. Pryadko was supported in part by ARO Grant W911NF-14-1-0272 and NSF Grant PHY-1416578. The work of A.A. Kovalev was supported in part by NSF Grant PHY-1415600.

REFERENCES


