Sparse Representations for Codes and
the Hardness of Decoding LDPC Codes

Nandakishore Santhi∗
Theoretical Division, CCS Division and the Center for Non Linear Studies.
LANL, PO 1663, MS B213, Los Alamos, NM 87545.
n.santhi@lanl.gov

Abstract—The Maximum Likelihood Decoding problem is known to be NP-hard for binary linear codes, while belief propagation decoding is known to work well in practice for several LDPC codes. In this paper we give a polynomial time reduction from the Maximum Likelihood Decoding (MLD) problem for binary linear codes to the Weighted MLD problem for (3,3)-LDPC codes. The reduction proves the NP-hardness of Weighted MLD for (3,3)-LDPC codes. It also provides a method which can be used to transform the decoding problem for dense codes to the decoding of sparse codes. The later problem is often more amenable to the use of belief propagation algorithm. For ease of presentation, we have organized the total reduction in several intermediate reductions, most of which are elementary and easy to follow.

I. INTRODUCTION

We consider the MLD problem for general binary linear codes. We give a series of polynomial time reductions which transform this hard problem to the weighted MLD of sparse codes. The parity check matrix of the resulting sparse codes have row and column weights \( \leq 3 \).

After the preparation of this note, we became aware of the results of McGregor and Milenkovic [5] and Xu and Hassibi [6]. Our reduction technique is completely independent of and totally different from the aforementioned results. In [5], the hardness of approximating stopping and trapping sets in LDPC codes is considered. The following is also proved:

Theorem 1 (MaxLD-LDPC is NP-hard[5]) Maximum Likelihood Decoding of LDPC codes with parity-check matrices of column weight 3 and row weight bounded from above by some (unspecified) constant \( \ell \geq 3 \) is NP-hard.

In [6], the authors show that the exact ML-decoding of certain expander graph based LDPC codes is NP-hard. The objective of our polynomial reduction on the other hand is entirely different: we transform the MLD of any binary linear code to the weighted MLD of some sparse code with both column and row weights bounded from above by 3. Thus our reduction could be directly employed in a preprocessing step to transform decoding of dense codes to that of related sparse codes. This can potentially be advantageous in certain instances as dense parity check matrices usually do not do well with the computationally efficient belief propagation algorithm.

II. POLYNOMIAL REDUCTION FROM MLD OF GENERAL LINEAR CODES TO MLD OF (3,3)-LDPC CODES

In this section we give a polynomial time reduction from MLD to MLD-(3,3)-LDPC. That is, we show that every instance of Maximum Likelihood Decoding of a linear code \( \mathbb{C} \) can be reduced to the Maximum Likelihood Decoding problem for a (3,3)-LDPC code. Here, by an \((r,c)\)-LDPC, we mean a binary linear code with a generalized (redundant) low density parity check matrix having row weight \( \leq r \) and column weight \( \leq c \).

The total reduction proceeds through several intermediate reductions – some steps reduce to classical hard problems found for example in Garey and Johnson [2], while some steps have been constructed to make full use of the structure in the problem. Our reduction is shown schematically in Figure 1. To simplify the overall reduction, we give intermediate reductions from MLD to various well known problems – that is, solving any of these problems in polynomial time implies a polynomial solution to the MLD problem. We rely on the specific form of the weighted MAX-CUT graph which results after our reduction from the MLD problem – this aspect will be clear in the subsection which deals with this part of the reduction.

MLD is a well known NP-hard problem. The corresponding decision problem was shown to be NP-complete by a reduction from THREE-DIMENSIONAL-MATCHING by Berlekamp, McEliece and van Tilborg in [1]:

MLD Problem: Maximum Likelihood Decoding of Linear Codes

Instance: An \( m \times n \) matrix \( \underline{H} \) over \( \mathbb{F}_2 \), a target vector \( \underline{\gamma} \in \mathbb{F}_2^m \), and an integer \( W_{ML} > 0 \).

Question: Is there a vector \( \underline{v} \in \mathbb{F}_2^n \) of Hamming weight \( \leq W_{ML} \) such that \( \underline{H}\underline{v}^T = \underline{\gamma}^T \)?
In what follows, a step by step polynomial time reduction from MLD to MLD-(3,3)-LDPC is given. We demonstrate the hardness of the search problem associated with MLD-(3,3)-LDPC by showing that the corresponding decision problem is NP-complete. This will be accomplished by a polynomial time reduction from the decision problem corresponding to MLD.

A. Maximum Likelihood Decoding as an instance of the Weighted Maximum XOR Satisfiability problem

We convert the MLD problem to an instance of a WEIGHTED-MAX-XORSAT problem with polynomial effort. The decision version of the later problem is:

WEIGHTED-MAX-XORSAT

Problem: Maximum Weighted XOR Satisfiability

Instance: A set \( \mathcal{X} \) of variables, a collection \( \mathcal{C}_X \) of XOR clauses over \( \mathcal{X} \), a set of positive integer weights \( w_j \) corresponding to each clause \( C_j \in \mathcal{C}_X \) and a non-negative integer \( W_X \leq \sum_j w_j \).

Question: Is there a truth assignment for \( \mathcal{X} \) such that the total weight of the satisfied clauses is at least \( W_X \)?

**Theorem 2** \( \text{MLD} \leq_p \text{WEIGHTED-MAX-XORSAT} \)

**Proof:** Every row of the matrix condition \( H_\mathcal{X}^T = \mathcal{S}^T \) is in one of the two forms:

\[
\sum_j h_{ij}v_j = 0 \quad \text{if} \quad s_i = 0
\]

\[
\sum_j h_{ij}v_j = 1 \quad \text{if} \quad s_i = 1
\]

where the sum is over \( \mathcal{P}_2 \) and corresponds to XOR. If \( s_i = 0 \), then by negating the last \( v_j \) which participates in the \( i \)th parity check, we can convert the conditions to a common form:

\[
\sum_j \ell_j = 1
\]

where \( \ell_j \) denotes a literal (either \( v_j \) or a negated \( \neg v_j \)). The set of \( m \) such XOR clauses we call the code clauses, denoted \( \mathcal{X}_c \). Similarly form \( n \) simple clauses, each consisting of the single literal \( \neg v_j : j \in \{1,2,\ldots,n\} \) and call this set the radius clauses, \( \mathcal{S}_c \).

Let \( \mathcal{C}_X = \mathcal{C}_X \cup \mathcal{S}_c \) be the set of XOR clauses. Assign a weight of \( (n+1) \) to each of the code clauses \( \mathcal{C}_X \) and a weight of \( 1 \) to each of the radius clauses, \( \mathcal{S}_c \). Every word \( \mathcal{g}_c \) such that \( H_\mathcal{C}_X^T = \mathcal{g}_c^T \) satisfies all the \( m \) code clauses. There are \( 2^{v_m} \) such words for every target vector \( \mathcal{g} \). In addition an MLD word \( \mathcal{g} \) also satisfies at least \( (n-W_{ML}) \) of the radius clauses. The total weight of the satisfied clauses is therefore at least \( m(n+1) + (n-W_{ML}) \) for the MLD word \( \mathcal{g} \). On the other hand, let us consider a non-MLD word \( \tilde{\mathcal{g}} \). Such a word will either satisfy the matrix condition \( H_\mathcal{C}_X^T = \tilde{\mathcal{g}}^T \) or not. If it does satisfy the matrix condition, then it satisfies less than \( (n-W_{ML}) \) of \( \mathcal{S}_c \).

If it does not satisfy the matrix condition, then it satisfies at least \( (m-1) \) of \( \mathcal{X}_c \). In both cases, the word \( \tilde{\mathcal{g}} \) satisfies a total weight less than \( m(n+1) + (n-W_{ML}) \). Thus, the set of variables \( \mathcal{X}_X = \{v_j : j \in \{1,2,\ldots,n\}\} \), the set of clauses \( \mathcal{C}_X \), the set of weights \( w_j \) associated with the clause \( \mathcal{C}_X \) and the target weight of \( W_X = m(n+1) + (n-W_{ML}) \) together form an instance of a WEIGHTED-MAX-XORSAT problem, solving which solves the MLD problem.

**B. Maximum Likelihood Decoding as an instance of the Weighted Maximum 3-XOR Satisfiability problem**

We transform MLD problem to an instance of the WEIGHTED-MAX-3XORSAT problem, the decision version of which is as follows:

**WEIGHTED-MAX-3XORSAT**

**Problem:** Maximum Weighted 3-XOR Satisfiability

**Instance:** A set \( \mathcal{X}_3 \) of variables, a collection \( \mathcal{C}_3 \) of XOR clauses over \( \mathcal{X}_3 \) such that each clause has \( \leq 3 \) literals, a set of positive integer weights \( w_j \) corresponding to each clause \( C_j \in \mathcal{C}_3 \) and a non-negative integer \( W_{3X} \leq \sum_j w_j \).

**Question:** Is there a truth assignment for \( \mathcal{X}_3 \) such that the total weight of the satisfied clauses is at least \( W_{3X} \)?

**Proposition 1** Let \( C_3 = \{ \ell_1 + \ell_2 + \cdots + \ell_t \} \). Then there exists a truth assignment which satisfies \( C_3 \) iff there exists an auxiliary variable \( u \) and a truth assignment such that both the clauses \( C_1 \) and \( C_2 \) are satisfied, where \( C_1 = \{ \ell_1 + \ell_2 + \cdots + \ell_{t-2} + u \} \) and \( C_2 = \{ \ell_1 - u + \ell_{t-1} + \ell_t \} \).

**Proof:** \( C_3 = 1 \) iff either \( (\ell_1 + \ell_2 + \cdots + \ell_{t-2}) \land (\ell_{t-1} + \ell_t) \) or \( (\ell_1 + \ell_2 + \cdots + \ell_{t-2}) \land (\ell_{t-1} + \ell_t) \). In the first case, setting \( u = 0 \), and using the truth assignment for the literals \( \ell_j \) which satisfies \( C_3 \), both \( C_1 \) and \( C_2 \) are satisfied, while setting \( u = 1 \) satisfies neither \( C_1 \) nor \( C_2 \). In the second case, reversing the assignment for \( u \) has the same effect. Conversely, if setting \( u = 0 \) satisfies both \( C_1 \) and \( C_2 \), then this must imply the first case and if setting \( u = 1 \) satisfies both \( C_1 \) and \( C_2 \), then this must imply the second case.

**Theorem 3** \( \text{MLD} \leq_p \text{WEIGHTED-MAX-3XORSAT} \)

**Proof:** Let the weight of the \( i \)th parity check row be \( w(h_i) \).

If \( w(h_i) = 2 \) we do not have to add any auxiliary variables for that parity check. Else if \( w(h_i) > 2 \), applying Proposition 1 and using exactly \( w(h_i) - 3 \) auxiliary variables \( u_{hi} \), we can convert the \( i \)th code clause into a conjunction of 3-XOR code clauses. Assign each of the clauses in this conjunction a weight of \( (n+1) \) and pool all the 3-XOR code clauses belonging to the different parity checks together to form a collection of 3-XOR code clauses, \( \mathcal{X}_3 \). Let \( \mathcal{C}_3 = \mathcal{C}_3 \cup \mathcal{S} \) and \( \mathcal{C}_X = \mathcal{X}_3 \cup \mathcal{S} \), where \( S \) is the set of all auxiliary variables introduced. A truth assignment satisfying the maximum weight for the original WEIGHTED-MAX-XORSAT problem now corresponds to a maximum weight \( W_{3X} = |\mathcal{C}_X|/(n+1) \) satisfying truth assignment for the WEIGHTED-MAX-3XORSAT problem with a variable set \( \mathcal{X}_3 \). Also, \( |\mathcal{C}_X| = \Theta(w(H) + n) \) and \( |\mathcal{C}_X| = \Theta(w(H) + n) \).

**C. Maximum Likelihood Decoding as an instance of the Weighted Maximum 3-XOR Satisfiability problem**

The decision version of WEIGHTED-MAX-3SAT is:
WEIGHTED-MAX-3SAT
Problem: Maximum Weighted 3-OR Satisfiability
Instance: A set $U_{3S}$ of variables, a collection $C_{3S}$ of OR clauses over $U_{3S}$ such that each clause has $\leq 3$ literals, a set of positive integer weights $w_j$ corresponding to each clause $C_j \in C_{3S}$ and a non-negative integer $W_{3S} \leq \sum_j w_j$.
Question: Is there a truth assignment for $U_{3S}$ such that the total weight of the satisfied clauses is at least $W_{3S}$?

The following two results are elementary:

Proposition 2 (a) A truth assignment satisfies $C \overset{\text{def}}{=} (\alpha \lor \beta + \gamma)$ iff it satisfies $C_1$, $C_2$, $C_3$ and $C_4$, where:

$C_1 \overset{\text{def}}{=} (\alpha \lor \beta \lor \gamma), C_2 \overset{\text{def}}{=} (\alpha \lor \neg \beta \lor \neg \gamma),$
$C_3 \overset{\text{def}}{=} (\neg \alpha \lor \beta \lor \neg \gamma), C_4 \overset{\text{def}}{=} (\neg \alpha \lor \neg \beta \lor \gamma)$

(b) A truth assignment satisfies $C \overset{\text{def}}{=} (\alpha + \beta)$ iff it satisfies $C_1$ and $C_2$ where:

$C_1 \overset{\text{def}}{=} (\alpha \lor \beta), C_2 \overset{\text{def}}{=} (\neg \alpha \lor \neg \beta)$

Now MLD can be transformed into WEIGHTED-MAX-3SAT:

Theorem 4 MLD $\leq P$ WEIGHTED-MAX-3SAT

Proof: Substitute every 3-XOR clause in $X_{3S}$ of weight $(n+1)$ with a set of four 3-OR clauses each of weight $(n+1)$ and every 2-XOR clause of weight $(n+1)$ with two 2-OR clauses each of weight $(n+1)$ using Proposition 2. Let this new set of 3-OR clause be $X_{3S}$. Now let $C_{3S} \overset{\text{def}}{=} X_{3S} \cup R$.]

D. Maximum Likelihood Decoding as an instance of the Weighted Maximum 2-OR Satisfiability problem

We will now transform the MLD problem into an instance of WEIGHTED-MAX-2SAT. This well known problem has the following decision version:

WEIGHTED-MAX-2SAT
Problem: Maximum Weighted 2-OR Satisfiability
Instance: A set $U_{2S}$ of variables, a collection $C_{2S}$ of OR clauses over $U_{2S}$ such that each clause has $\leq 2$ literals, a set of positive integer weights $w_j$ corresponding to each clause $C_j \in C_{2S}$ and a non-negative integer $W_{2S} \leq \sum_j w_j$.
Question: Is there a truth assignment for $U_{2S}$ such that the total weight of the satisfied clauses is at least $W_{2S}$?

Trevisan, Sorkin, Sudan and Williamson [3] give the following gadget which may be verified using truth tables:

Proposition 3 A truth assignment satisfies $C \overset{\text{def}}{=} (\alpha \lor \beta \lor \gamma)$ iff it satisfies exactly 7 out of the following 8 2-OR clauses:

$(\alpha \lor \gamma), (\neg \alpha \lor \gamma), (\alpha \lor \neg \gamma), (\neg \alpha \lor \neg \gamma),$
$(\gamma \lor \neg \eta), (\neg \gamma \lor \eta), (\beta \lor \eta), (\beta \lor \eta)$

where $\eta$ is an auxiliary variable. Moreover, any assignment which fails to satisfy $C$ satisfies $\leq 5$ of these 2-OR clauses.

Theorem 5 MLD $\leq P$ WEIGHTED-MAX-2SAT

Proof: Earlier we reduced MLD to an instance of WEIGHTED-MAX-3SAT. Any 2-OR code clauses $(\alpha \lor \beta)$, are first converted to a 3-OR form $(\alpha \lor \beta \lor \gamma)$. Now each of the 3-OR code clauses $C_i$ are replaced with eight 2-OR clauses as in Proposition 3 with a weight of $(n+1)$ each. This introduces an auxiliary variable $\eta_j$ each time. Let this set of 2-OR code clauses be $X_{2S}$ and let $C_{2S} \overset{\text{def}}{=} X_{2S} \cup R$. The clauses $C_{2S}$ and the corresponding weights so obtained along with the target weight of $W_{2S} = |X_{2S}|(n+1) + (n-W_{ML})$ is an instance of WEIGHTED-MAX-2SAT, which solves MLD.

E. Maximum Likelihood Decoding as an instance of the Multi-Graph Maximum Cut problem

Any graph with positive integer edge weights can be considered a multi-graph with unit edge weights and vice-versa. Therefore we interchangeably use a graph with positive integer edge weights and its corresponding multi-graph. We transform the MLD to an instance of the MAX-CUT problem, the decision version of which is:

MAX-CUT
Problem: Maximum 2-way Multi-Cut
Instance: A multi-graph $G = (V, E)$ and a non-negative integer $K \leq |E|$.
Question: Is there a partition of $V$ into disjoint subsets $V_1$ and $V_2$ such that the subset of edges in $E$ with one vertex in $V_1$ and the other vertex in $V_2$ has cardinality at least $K$?

Theorem 6 MLD $\leq P$ MAX-CUT

Proof: Earlier we transformed MLD to an instance of the WEIGHTED-MAX-2SAT problem. Now we take this instance and convert it into a multi-graph max-cut problem. We construct a multi-graph conforming to the set of 2-XOR clauses $X_{2S}$ with integer weights assigned to each clause. The graph construction is similar to the one by McCormick, Rao and Rinaldi [4].

For every variable appearing in various literals within the clauses $C_i \in C_{2S}$, we create two nodes, one for the variable and the other for the negated version of the variable. We also add $3 \sum_{j=1}^{|X_{2S}|} w_j$ edges between each of these node pairs, where $w_j$ is the positive integer weight of clause $C_i$.

Now introduce a special node $F$ (meaning logic FALSE). For every 2-OR clause $C_i \in C_{2S}$, such that $C_i \overset{\text{def}}{=} (\ell_1 \lor \ell_2)$, we add a sub-graph connecting $(\ell_1, \ell_2), (\ell_1, F)$ and $(\ell_2, F)$ with $w_j$ edges each. For every trivial one-literal clause $C_i \in C_{2S}$ such that $C_i = \ell$, we construct a sub-graph by adding $2w_j$ edges connecting $(\ell, F)$. See Figure 2 for a depiction of the multi-graph construction. Now a maximum cut in this graph has $K = 2W_{2S} + 3|X_{2S}| \cdot (\sum_{j=1}^{|X_{2S}|} w_j)$ edges and corresponds to a truth assignment to the variables in $X_{2S}$ which maximizes the weight of the satisfied 2-OR clauses in $X_{2S}$.

Indeed the max-cut corresponds to the maximum-weight 2-OR assignment – when a 2-OR clause is satisfied, at least one of the up to two associated literals are logic TRUE. Similarly any cut on $G$ will either cut the sub-graph corresponding to clause $C_i$ or not. If it does cut the sub-graph, then exactly $2w_j$
edges are added to the cut, otherwise no edges of the sub-graph are in the cut. A maximum cut will cut all the variable edges, contributing $3|\mathcal{E}_S| + (\sum_{i=1}^{n} w_i)$ edges to the cut. See Figure 3. In addition it will cut the maximum number of clause-sub-graphs which could be simultaneously satisfied thus having the largest total weight and adding an additional $2W_{2S}$ edges to the cut. Conversely, every such maximal cut induces a logically valid assignment to the variables in $\mathcal{Z}_S$ - assign logic value FALSE to all the literals in the vertex partition in the cut which includes the special node F. To all other literals assign logic value TRUE. This is a valid assignment as the nodes corresponding to a variable and its negation always appear in the two distinct partitions of $\mathcal{V}$. Furthermore, this particular variable assignment simultaneously satisfies a set of clauses with the maximum total weight. \(\blacksquare\)

\section*{F. Maximum Likelihood Decoding as an instance of Graph Cut Set Code MLD problem}

Let $w \in \mathbb{Q}_n$ be a vector of non-negative rational weights and let $d_w(\cdot, \cdot)$ denote the Hamming distance between two vectors in $\mathbb{F}_2^n$ weighted in co-ordinate $l$ with $w_l$. Also let $\mathbf{1}$ denote an all-ones vector of suitable dimension. The cutset code corresponding to a multi-graph $\mathcal{G}$ is denoted by $C(\mathcal{G})$ - a cutset code is the linear span of the incidence matrix of the graph. Cutset code maximum likelihood decoding is known to be NP-complete by a reduction from the MAX-CUT problem. Here we give a sketch of this reduction, in turn showing a reduction from MLD to MLD-CUTSETCODE. The decision version of the later problem is:

\textbf{MLD-CUTSETCODE}

\textbf{Problem:} Integer Weighted Maximum Likelihood Decoding of the Cut Set Code of a Multi-Graph

\textbf{Instance:} A multi-graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a vector $\ell \in \mathbb{F}_2^{\mathcal{|\mathcal{E}|}}$, a vector of integer weights $w \in \mathbb{Z}_n$ and an integer $W_{CS} \geq 0$.

\textbf{Question:} Is there a vector $\ell \in C(\mathcal{G})$ such that $d_w(\ell, 1) \leq W_{CS}$?

\textbf{Theorem 7} $\text{MLD} \leq_p \text{MLD-CUTSETCODE}$

\textbf{Proof:} Let the incidence matrix of the multi-graph constructed in Theorem 6 be $M$. The columns of $M$ correspond to the edges and the rows to the nodes. Associate with each column of $M$ the integer number of multiple edges with same end nodes. Let this vector of edge weights be $w$. Consider the cutset code which is the linear span of the rows of $M$. Finding the max-cut of the graph is clearly equivalent to finding the codeword $\ell$ in the cutset code such that $d_w(\ell, 1)$ is lowest. \(\blacksquare\)

\textbf{Remark:} The integer weighted Hamming distance can be replaced with the usual (unweighted) Hamming distance: simply replace each column $m_j$ in $M$ with $w_j$ copies. All other steps in the reduction remains the same.

\section*{G. Maximum Likelihood Decoding of a binary linear code as an instance of (3,3)-LDPC code MLD problem}

Let $\mathbf{0}$ denote an all-zeros vector of suitable dimension.

\textbf{MLD-(3,3)-LDPC}

\textbf{Problem:} Rational Weighted Maximum Likelihood Decoding of (3,3)-LDPC Codes

\textbf{Instance:} An $m \times n$ matrix $H'$ over $\mathbb{F}_2$ with column weight and row weight at most 3, a vector $\ell \in \mathbb{F}_2^m$, a vector of rational weights $w \in \mathbb{Q}_n$, and a rational $W_{LD} \geq 0$.

\textbf{Question:} Is there a vector $\ell \in \mathbb{F}_2^m$ such that $H'\ell^T = 0^T$ and $d_w(\ell, 2) \leq W_{LD}$?

\textbf{Theorem 8} $\text{MLD} \leq_p \text{MLD-(3,3)-LDPC}$

\textbf{Proof:} Let us consider the incidence matrix $M$ of Theorem 7. By construction, the special node $F$ is connected to every other literal node in the graph $\mathcal{G}$. Also the rows of $M$ are linearly dependent, since every column has exactly two 1’s. Arrange the rows of $M$ in such a manner that the node $F$ corresponds to the last row. By arranging the columns of $M$, it is possible to put it in the following form:

\[
\begin{bmatrix}
I & P \\
1 & b
\end{bmatrix}
\]

This means that the rows of $M$ excluding the last row forms a linearly independent generating set for $\mathcal{C}(\mathcal{G})$. So let $M' \overset{\text{def}}{=} [I | P]$ be the generator matrix obtained after removing the last
row of \( M \). Let \( H' \equiv \{ P^T \mid I \} \) be the corresponding systematic parity check matrix. The rows of \( H' \) have three 1’s each.

The columns of \( P^T \) have arbitrary number of 1’s. Let the number of 1’s of the \( l \)th column of \( P^T \) be \( c_l \). Also let the integer weight associated with this column in the metric \( d_w(\cdot, \cdot) \) be \( w_l \). If \( c_l > 3 \) we can replace the \( l \)th column with \( c_l - 2 \) columns with three 1’s each. The weight of each of the new columns in the weighted Hamming metric will be now \( \frac{w_l}{c_l - 2} \) which is rational. This also requires us to include \( c_l - 3 \) additional rows to enforce equality constraints for the newly introduced column variables. In the example of (2), the equality constraint is enforced by the new row marked with a *:

\[
\begin{bmatrix}
\vdots \\
1 \\
1 \\
\vdots \\
\end{bmatrix} \rightarrow \begin{bmatrix}
0 & 1 & 0 & 1 & 0 \\
\vdots \\
1 \\
1 \\
\vdots \\
\end{bmatrix}^* 
\]  

Since the original column had four 1’s, the weights associated with the newly formed columns are \( \frac{w_l}{2} \) each, where \( w_l \) was the weight associated with the original column.

Let \( H'' \) be the parity check matrix so obtained. Clearly the number of 1’s in each row and each column of \( H'' \) is \( \leq 3 \). Furthermore, the weighted maximum likelihood decoding of \( \frac{w_l}{2} \) w.r.t. \( H'' \) is equivalent to the maximum likelihood decoding of the dense parity check code.

Remarks:

(a) By construction, the columns of \( P^T \) have at most \( O(n) \) 1’s. Therefore, with a different weight assignment to the code-clauses, and a more careful analysis one can easily replace the weighted Hamming distance with the usual unweighted Hamming distance. However the length of the LDPC code so obtained is much larger, though still polynomial in \( n \).

(b) The LDPC matrix \( H'' \) which results after our reduction has \( O(w(H)) \) columns and \( O(w(H)) \) rows, where \( H \) is the parity check matrix of the original binary code.

The dimensions of the matrix \( H'' \) determine the complexity of belief propagation decoding of the sparse code.

III. CONCLUSIONS

We gave a polynomial time reduction which took the Maximum Likelihood Decoding (MLD) problem for binary linear codes to an equivalent Weighted Maximum Likelihood Decoding problem of \((3, 3)\)-LDPC codes. The reduction gives a proof for the NP-hardness of Weighted Maximum Likelihood Decoding of \((3, 3)\)-LDPC codes. While the Maximum Likelihood Decoding problem is known to be NP-hard for binary linear codes in general, belief propagation decoding is known to work well in practice for several LDPC codes.

In addition to showing the computational hardness of decoding \((3, 3)\)-LDPC codes, our reduction method provides a way to transform the MLD of dense codes to the decoding of sparse codes. We expect that the use of belief propagation decoding algorithm on the so formed sparse codes will be the subject of a future work.

ACKNOWLEDGMENT

Financial support in the form of LDRD ER/DR funds from the Los Alamos National Laboratory and the US Department of Energy (DoE) is gratefully acknowledged.

REFERENCES


